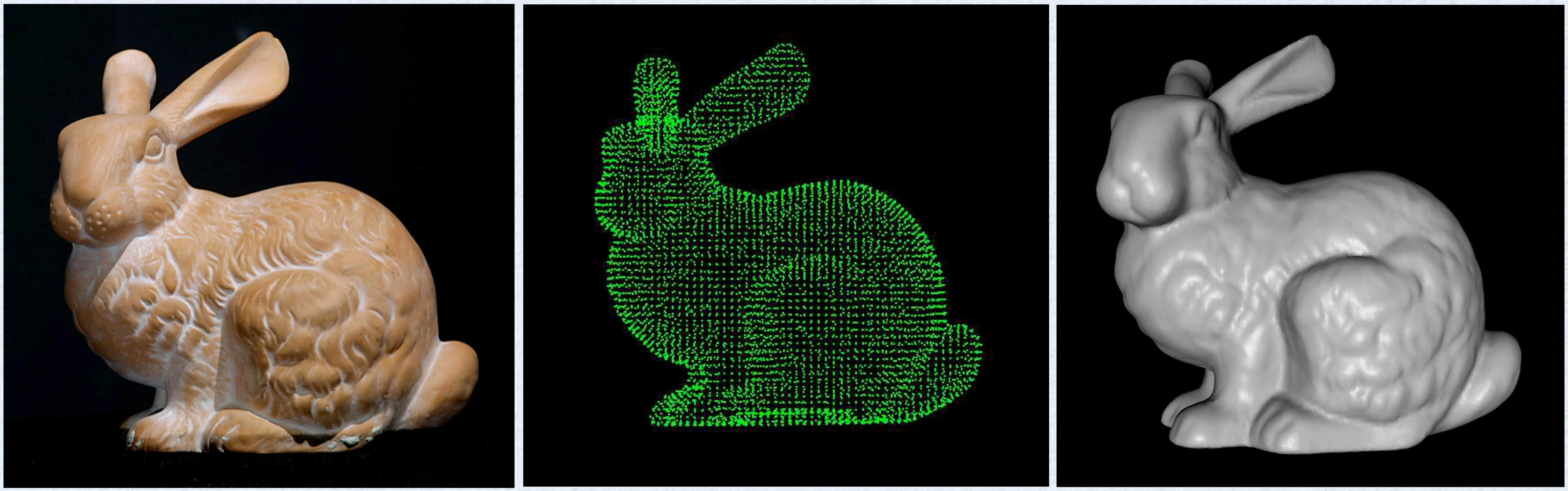
Geometric and Topological Guarantees for the WRAP Reconstruction Algorithm

Bardia Sadri

Joint Work Edgar Ramos

The Surface Reconstruction Problem



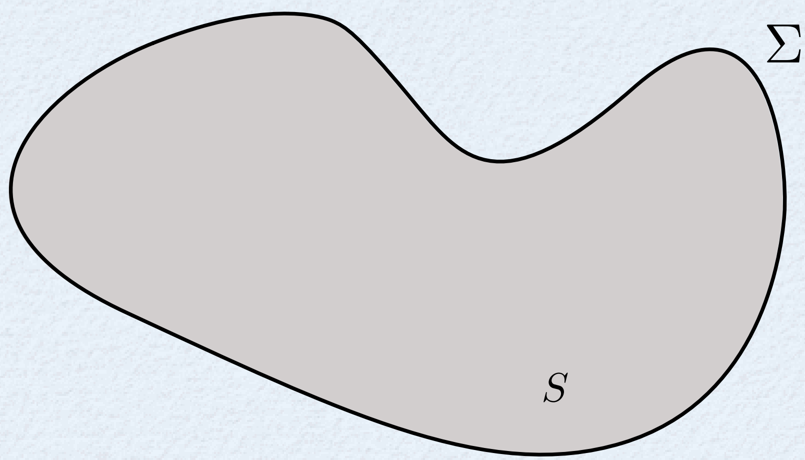
Given a point cloud sampled from a surface Σ, we want to compute a surface $\hat{\Sigma}$ that has the same topology as Σ and closely approximates it geometrically.

Shapes, Surfaces, and their Medial Axes

A shape is an open set S that has a "smooth" surface Σ for boundary.

Shapes, Surfaces, and their Medial Axes

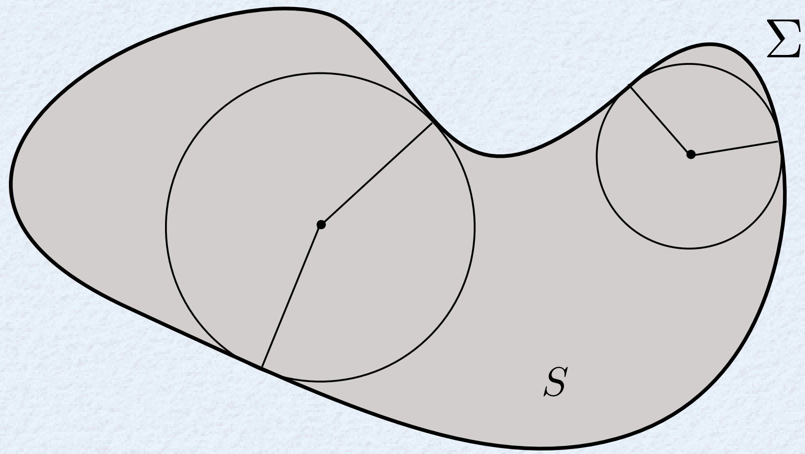
A shape is an open set S that has a "smooth" surface Σ for boundary.



Shapes, Surfaces, and their Medial Axes

A shape is an open set S that has a "smooth" surface Σ for boundary.

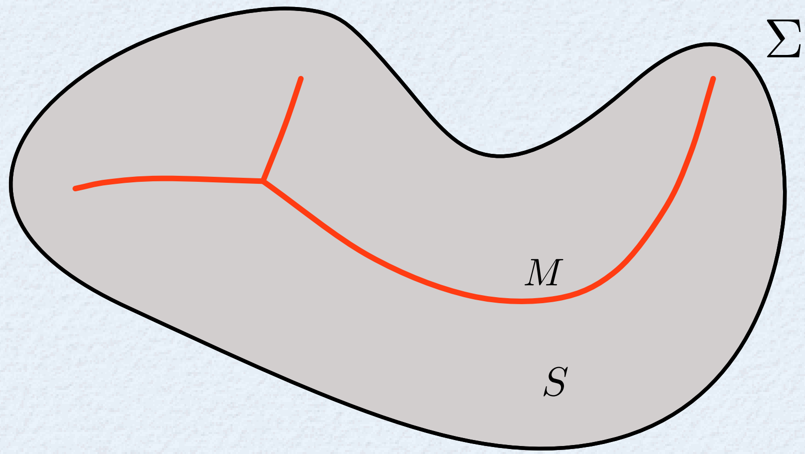
The medial axis (MA) of a shape S is the set of points in S that have ≥ 2 closest points in Σ.



Shapes, Surfaces, and their Medial Axes

A shape is an open set S that has a "smooth" surface Σ for boundary.

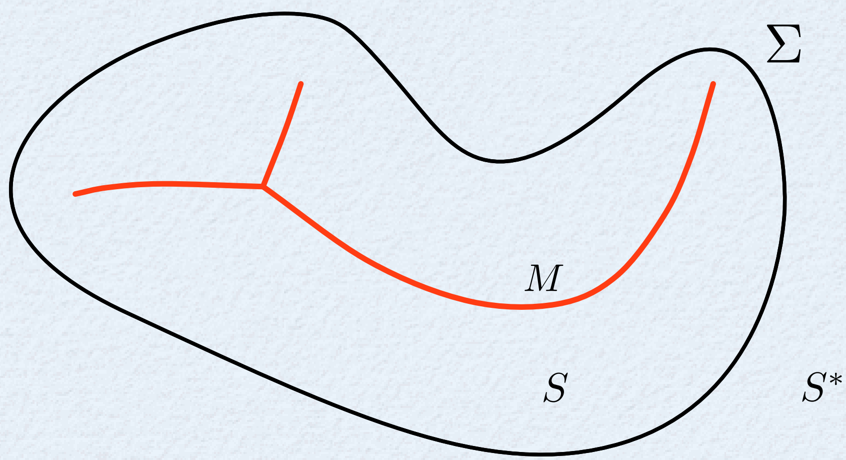
The medial axis (MA) of a shape S is the set of points in S that have ≥ 2 closest points in Σ.



Shapes, Surfaces, and their Medial Axes

A shape is an open set S that has a "smooth" surface Σ for boundary.

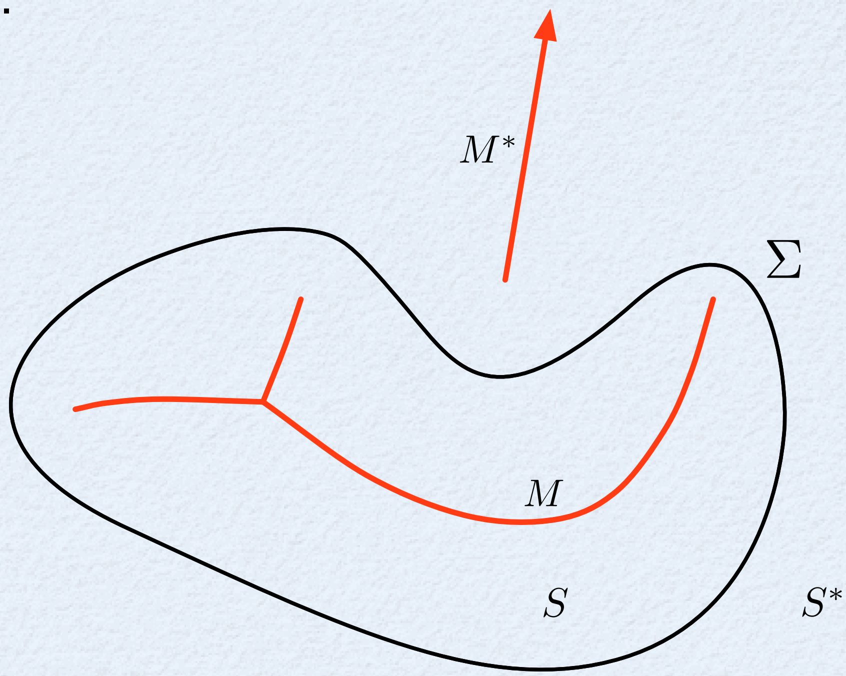
The medial axis (MA) of a shape S is the set of points in S that have ≥ 2 closest points in Σ.



Shapes, Surfaces, and their Medial Axes

A shape is an open set S that has a "smooth" surface Σ for boundary.

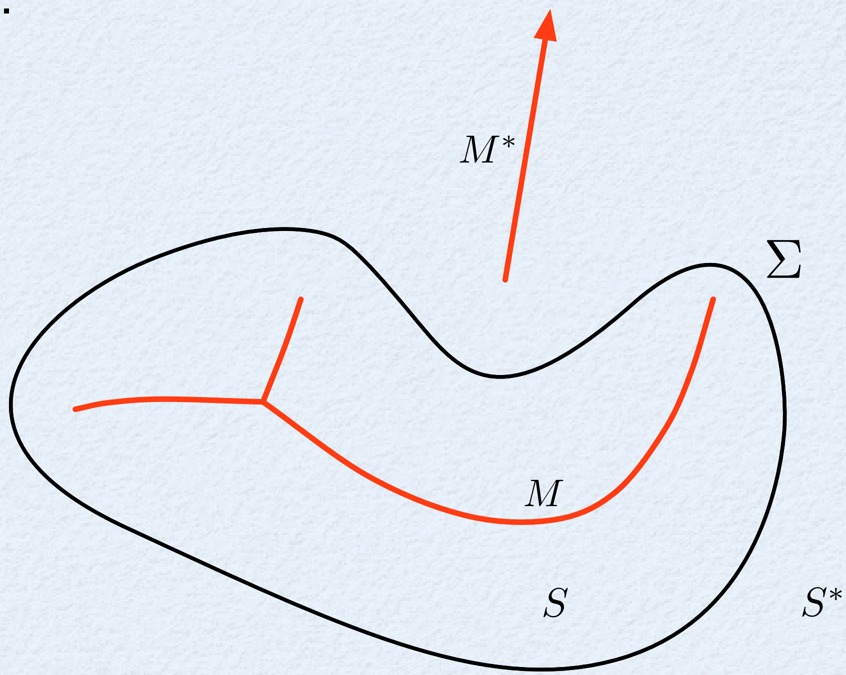
The medial axis (MA) of a shape S is the set of points in S that have ≥ 2 closest points in Σ.



Shapes, Surfaces, and their Medial Axes

A shape is an open set S that has a "smooth" surface Σ for boundary.

The medial axis (MA) of a shape S is the set of points in S that have ≥ 2 closest points in Σ.



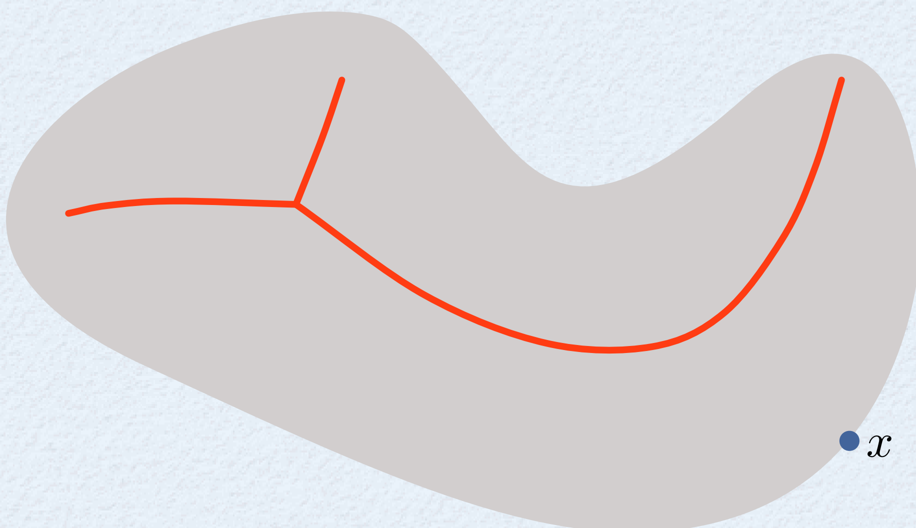
The medial axis of a surface Σ is the union of medial axes of all components of $\mathbb{R}^n \setminus \Sigma$.

Samples of Surfaces

We use the (relative) ε-sampling framework of [Amenta-Bern'99].

For a point $x \in \Sigma$, the local feature size of x is

$$\mathsf{lfs}(x) := d(x, M).$$



$P \subset \Sigma$ is an ε-sample if every $x \in \Sigma$ has a sample within distance $\varepsilon\,\mathsf{lfs}(x)$.

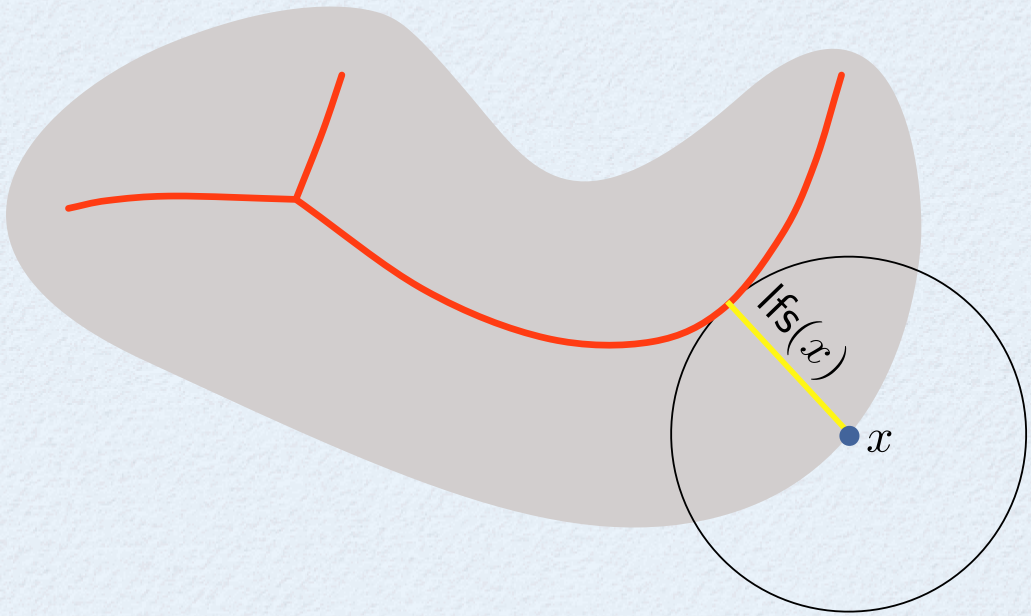
[Amenta-Bern'99] N. Amenta, M. Bern, Surface reconstruction by Voronoi filtering. Discrete & Computational Geometry, 1999.

Samples of Surfaces

We use the (relative) ε-sampling framework of [Amenta-Bern'99].

For a point $x \in \Sigma$, the local feature size of x is

$$\mathsf{lfs}(x) := d(x, M).$$



$P \subset \Sigma$ is an ε-sample if every $x \in \Sigma$ has a sample within distance $\varepsilon\,\mathsf{lfs}(x)$.

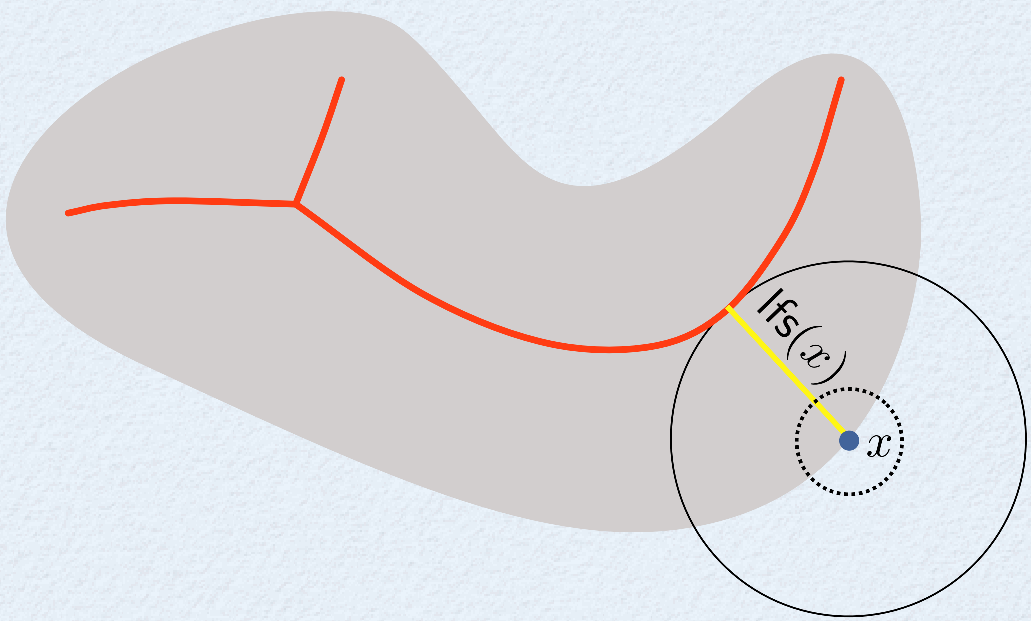
[Amenta-Bern'99] N. Amenta, M. Bern, Surface reconstruction by Voronoi filtering. Discrete & Computational Geometry, 1999.

Samples of Surfaces

We use the (relative) ε-sampling framework of [Amenta-Bern'99].

For a point $x \in \Sigma$, the local feature size of x is

$$\mathsf{lfs}(x) := d(x, M).$$



$P \subset \Sigma$ is an ε-sample if every $x \in \Sigma$ has a sample within distance $\varepsilon\,\mathsf{lfs}(x)$.

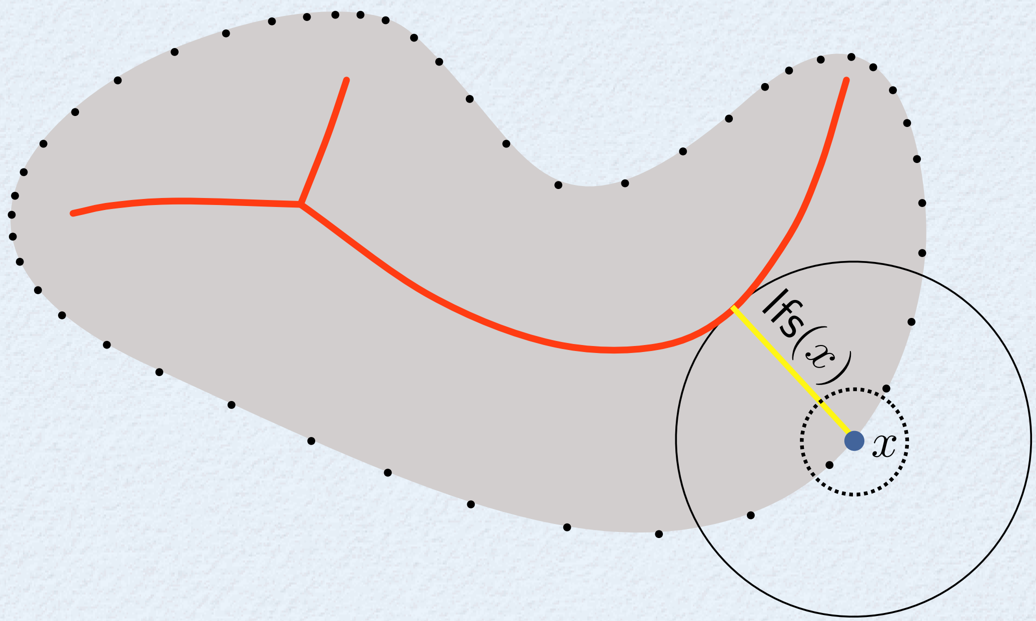
[Amenta-Bern'99] N. Amenta, M. Bern, Surface reconstruction by Voronoi filtering. Discrete & Computational Geometry, 1999.

Samples of Surfaces

We use the (relative) ε-sampling framework of [Amenta-Bern'99].

For a point $x \in \Sigma$, the local feature size of x is

$$\mathsf{lfs}(x) := d(x, M).$$



$P \subset \Sigma$ is an ε-sample if every $x \in \Sigma$ has a sample within distance $\varepsilon\,\mathsf{lfs}(x)$.

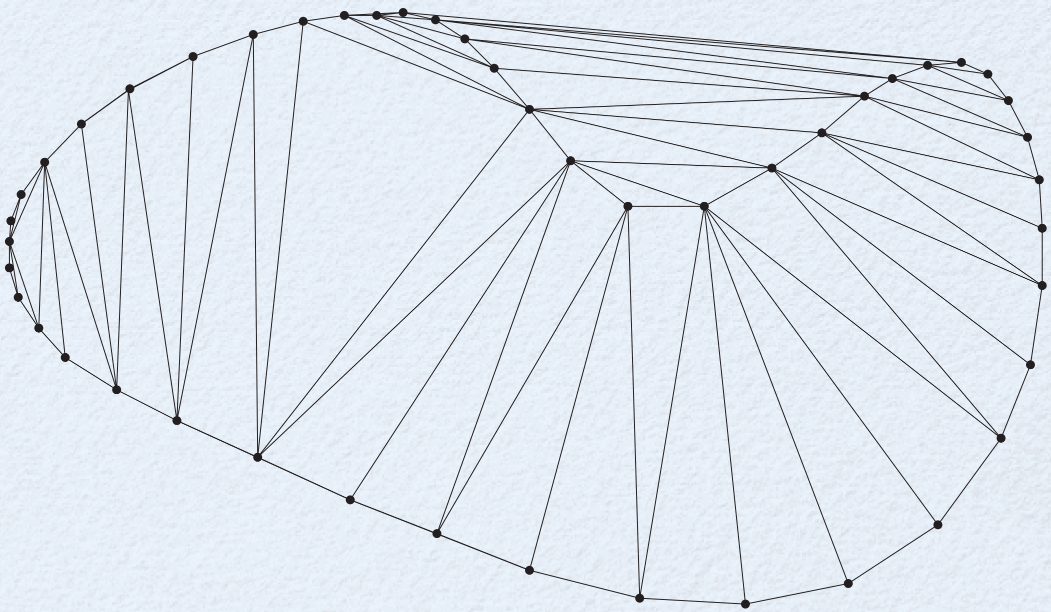
[Amenta-Bern'99] N. Amenta, M. Bern, Surface reconstruction by Voronoi filtering. Discrete & Computational Geometry, 1999.

There are many surface reconstruction methods!

- As 0-set of an approximate signed distance function: [Hoppe et al'92, Curless et al'96]

- As other iso-surfaces:
  NN Interpolation [Boissonnat-Cazals'02]
  MLS [Levin'98, Alexa et al'01, Amenta-Kil'04, Kolluri'05, Dey et al'05]
  SVM [Schölkopf et al'04]

- Delaunay Methods:
  [Boissonnat'84,
  Amenta-Bern'99,
  Amenta et al'91,
  Amenta-Choi-Kolluri'01]

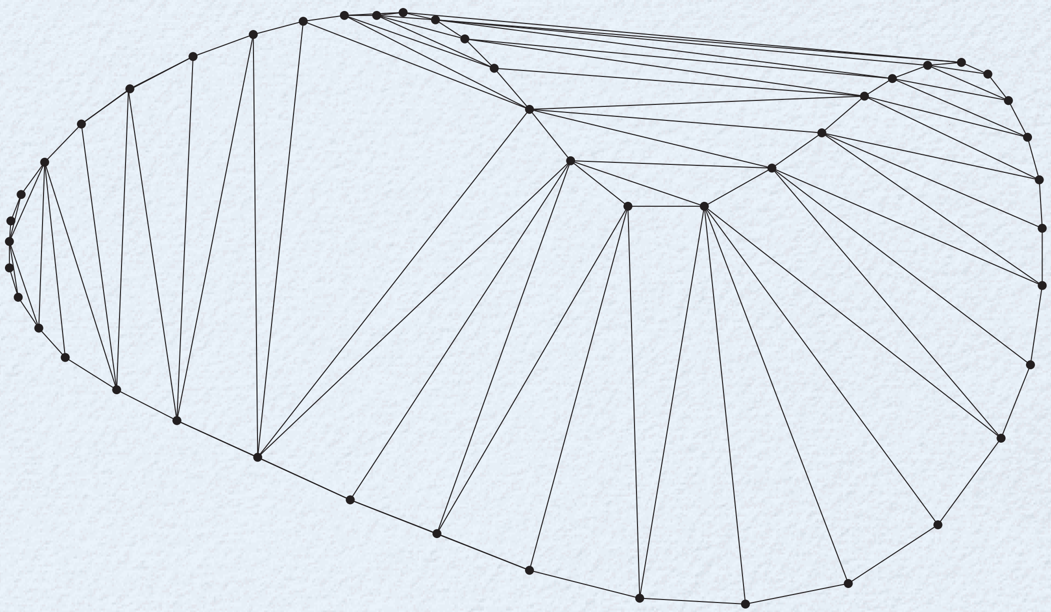


- Using distance functions:
  [Edelsbrunner'04, Chaine'03, Giesen-John'03, Dey et al'05]

There are many surface reconstruction methods!

- As 0-set of an approximate signed distance function: [Hoppe et al'92, Curless et al'96]

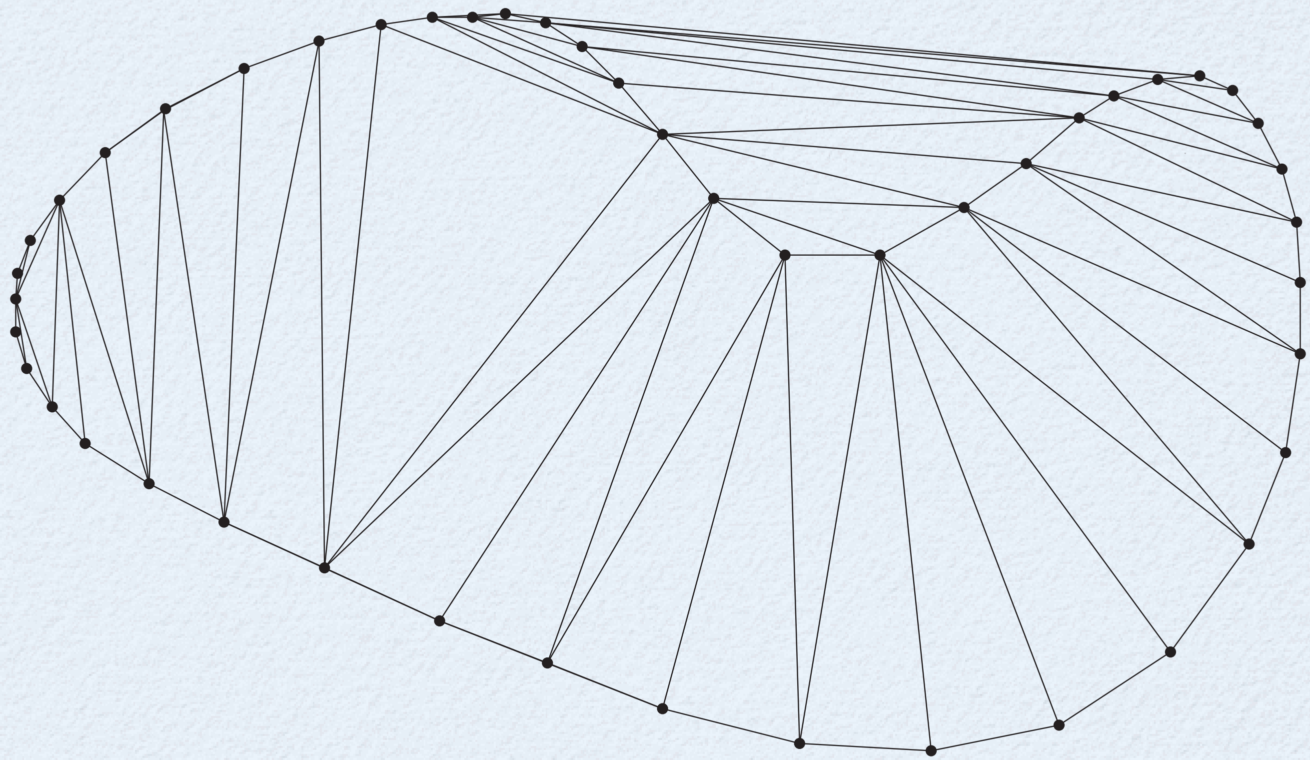
- As other iso-surfaces:
  NN Interpolation [Boissonnat-Cazals'02]
  MLS [Levin'98, Alexa et al'01, Amenta-Kil'04, Kolluri'05, Dey et al'05]
  SVM [Schölkopf et al'04]



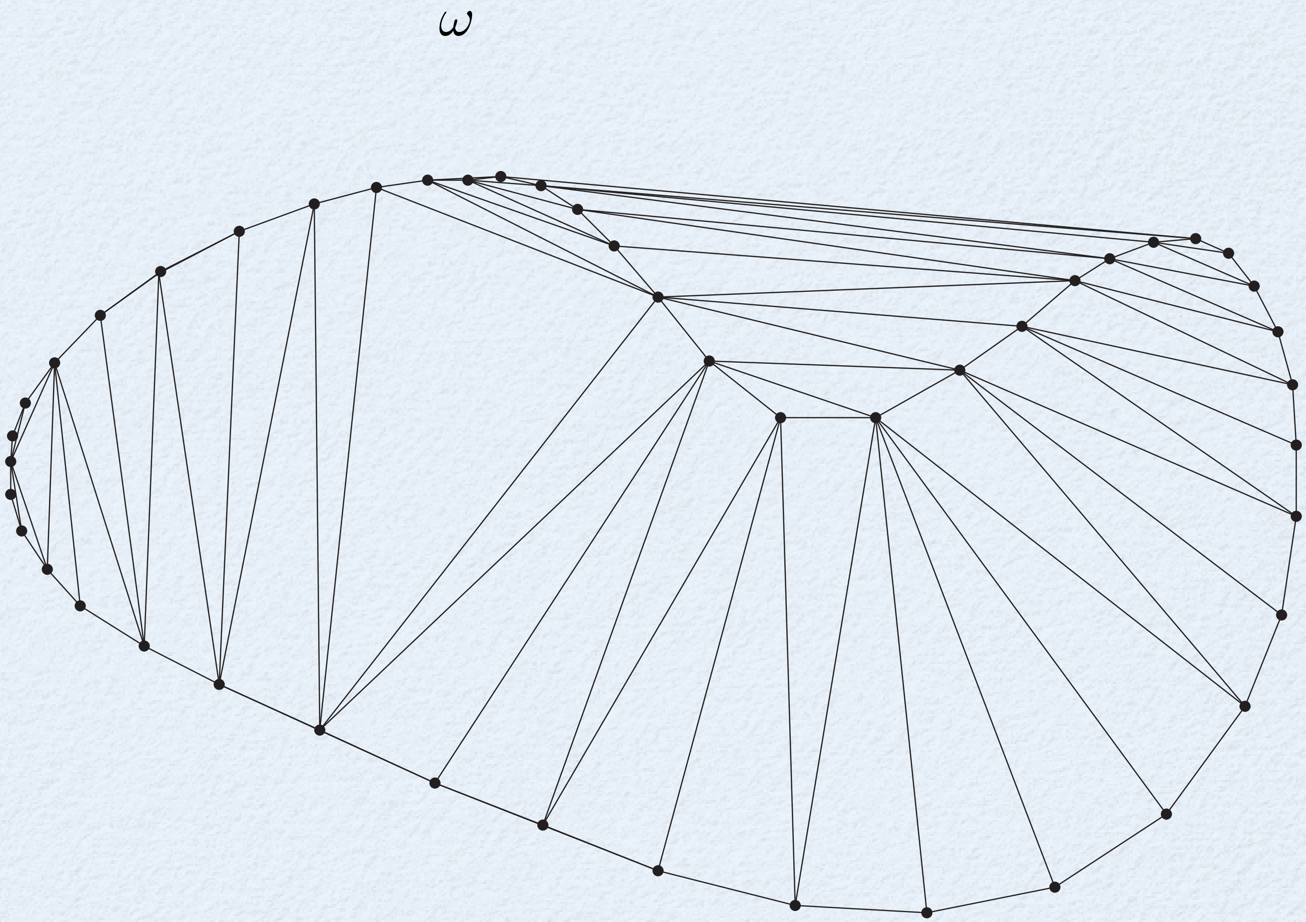
- Delaunay Methods:
  [Boissonnat'84,
  Amenta-Bern'99,
  Amenta et al'91,
  Amenta-Choi-Kolluri'01]

- Using distance functions:
  [Edelsbrunner'04, Chaine'03, Giesen-John'03, Dey et al'05]

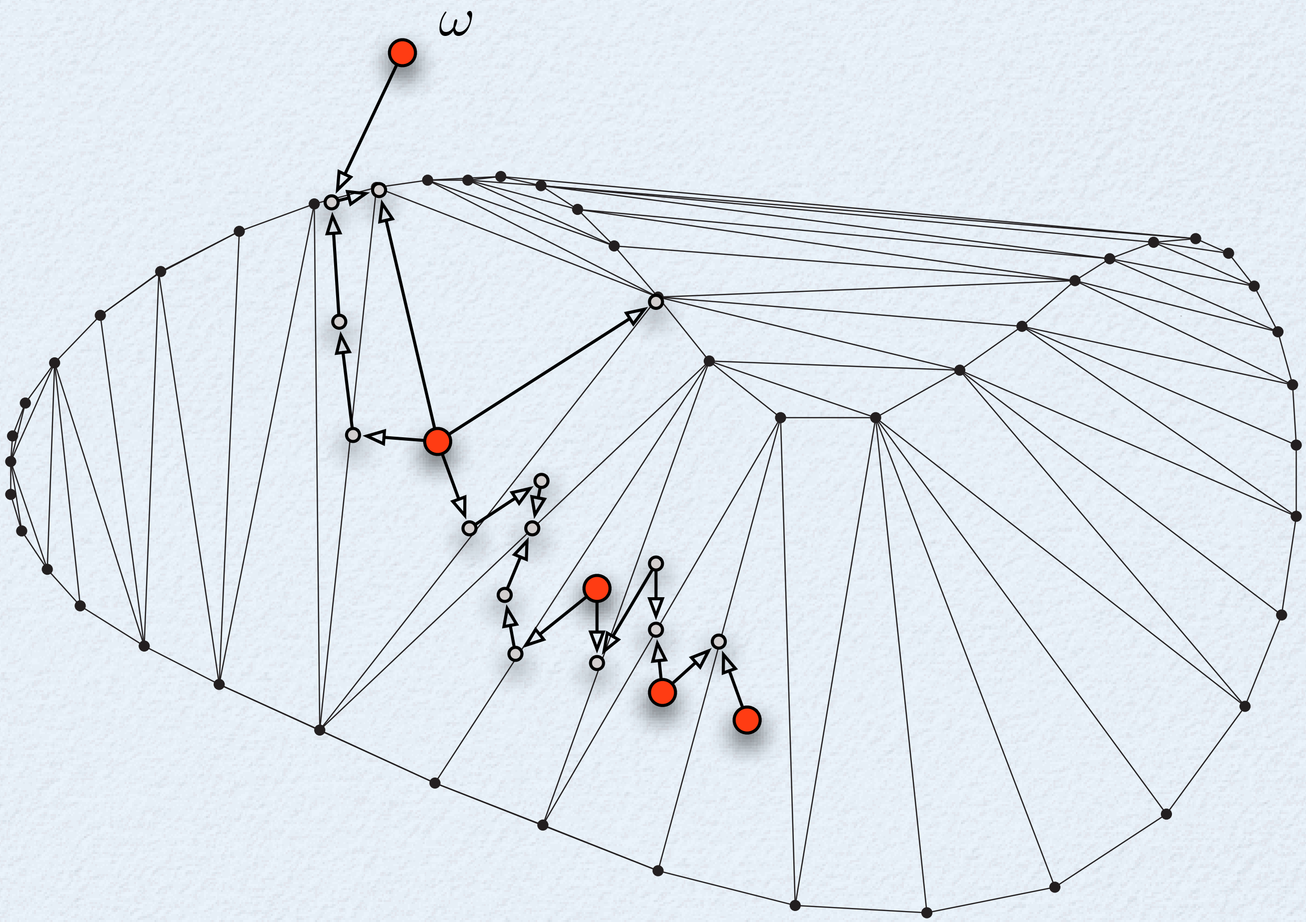
A Sketch of the WRAP Algorithm



A Sketch of the WRAP Algorithm

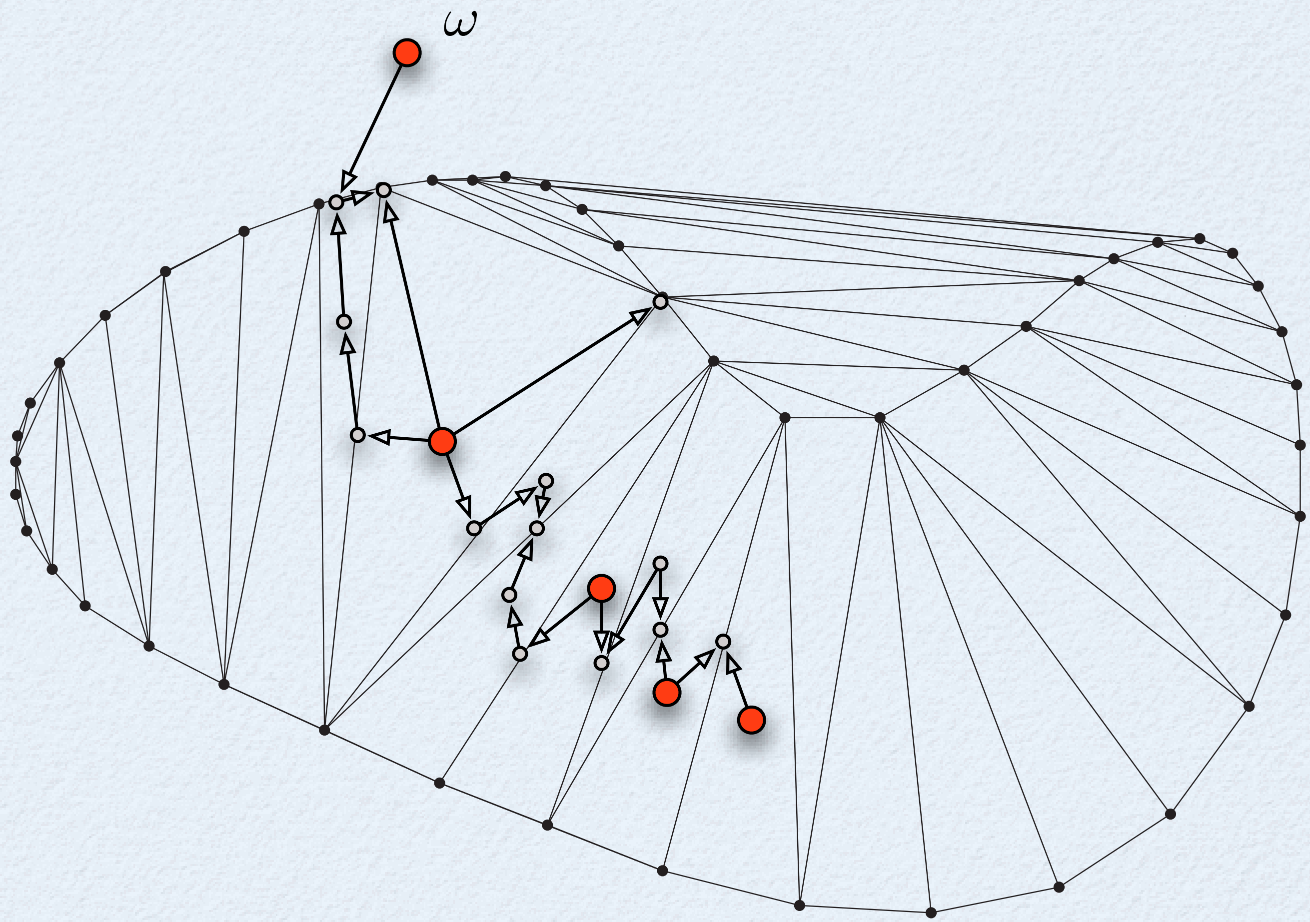


A Sketch of the WRAP Algorithm



A Sketch of the WRAP Algorithm

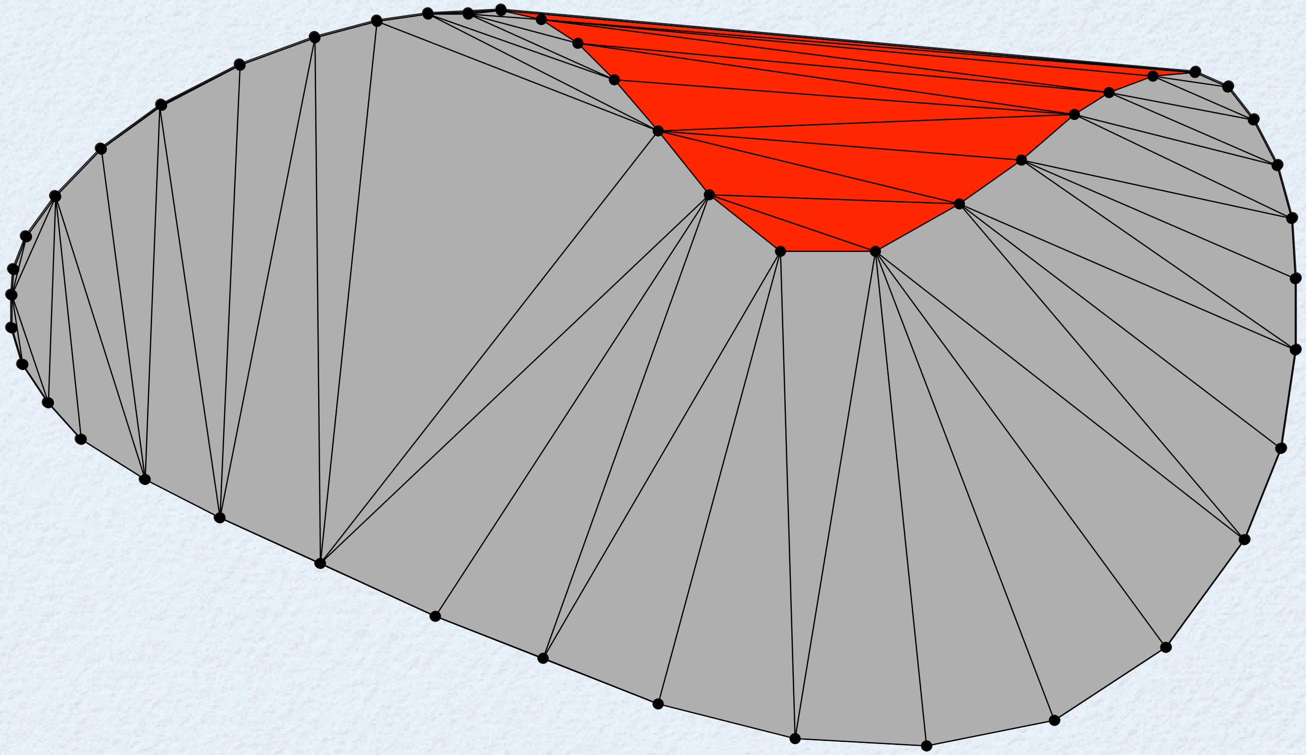
The WRAP Algorithm [Edelsbrunner'04]
For every $\tau \in \text{Del } P$, if τ is reachable from no "centered" simplex other than ω, then remove τ.



[Edelsbrunner'04] H. Edelsbrunner, Surface reconstruction by wrapping finite point-sets in space. Discrete & Computational Geometry, 2004.

A Sketch of the WRAP Algorithm

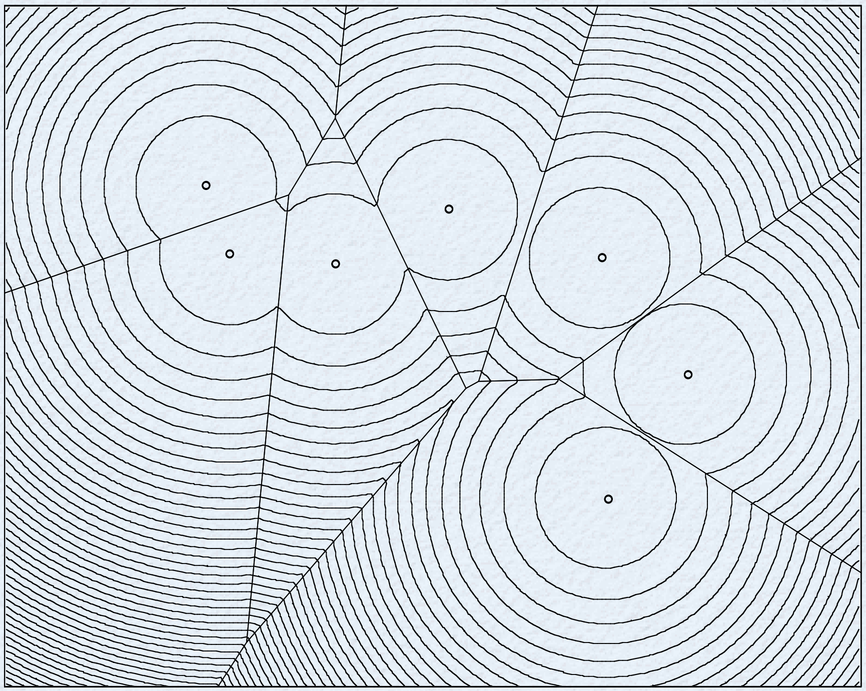
The WRAP Algorithm [Edelsbrunner'04]
For every $\tau \in \mathrm{Del}\ P$, if τ is reachable from no "centered" simplex other than ω, then remove τ.

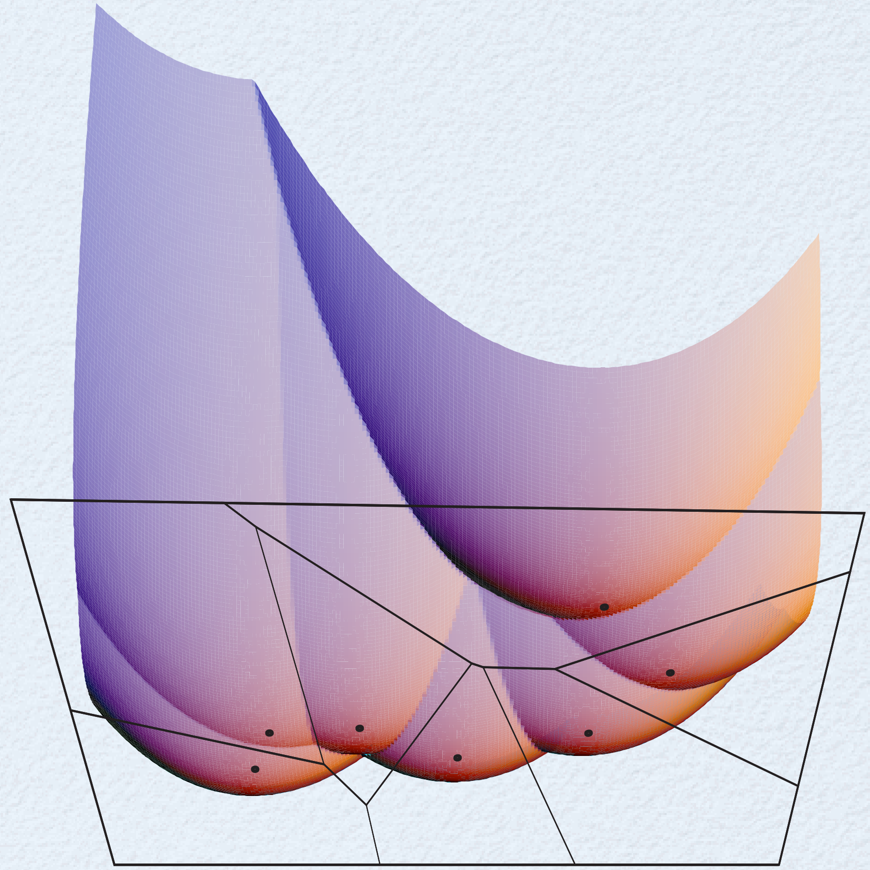


[Edelsbrunner'04] H. Edelsbrunner, Surface reconstruction by wrapping finite point-sets in space. Discrete & Computational Geometry, 2004.

The Machinery

(Squared) Distance to Discrete Point Sets



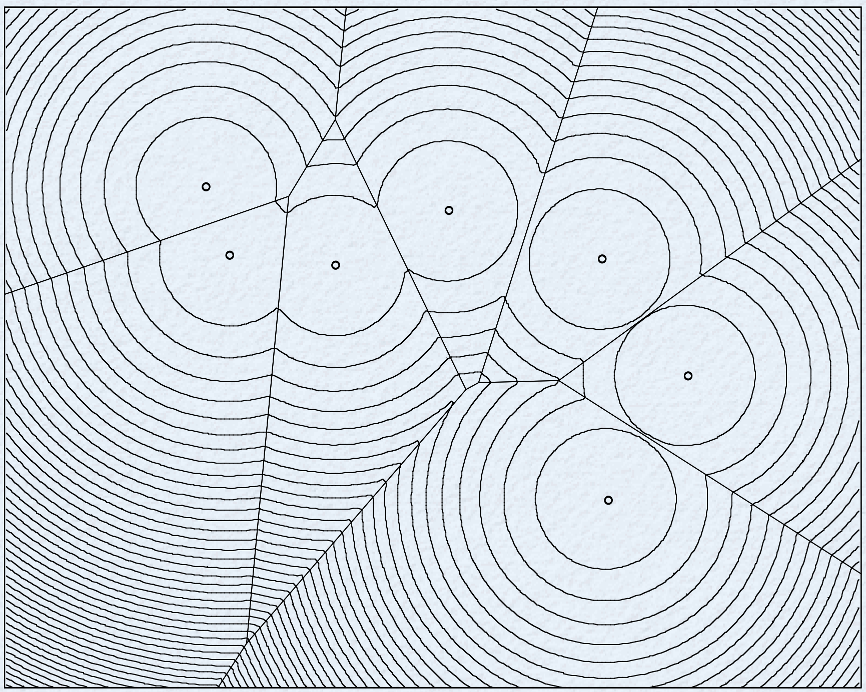


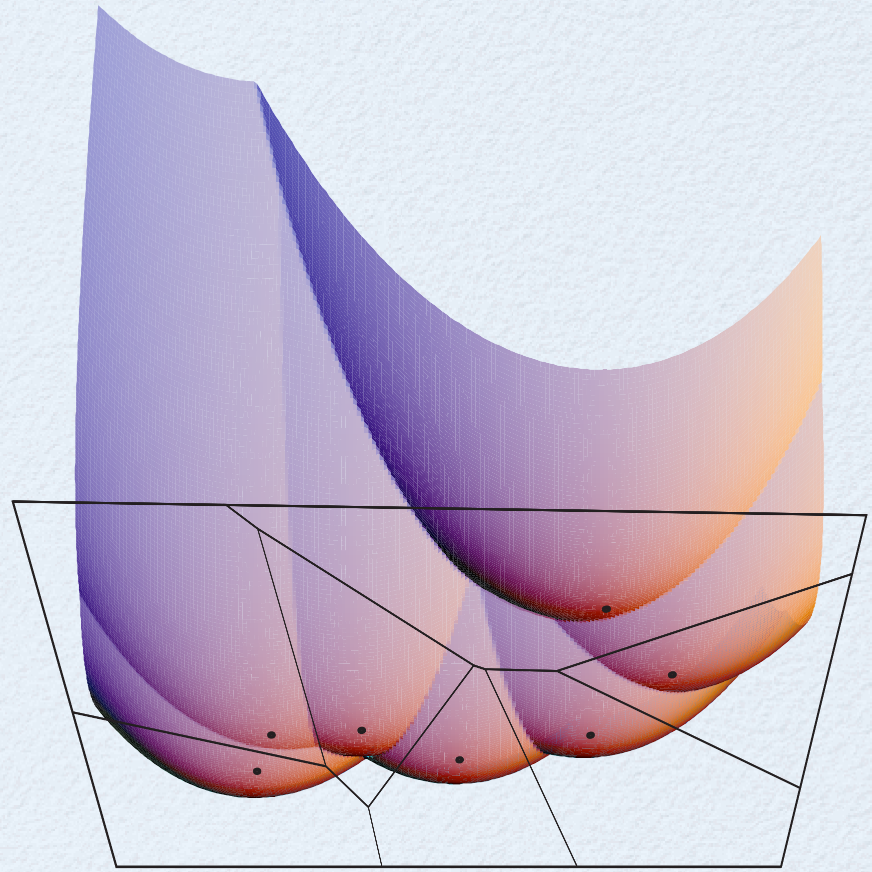
P is a discrete set of points

The squared distance function induced by P is

$$h(x) = \min_{p \in P} \|x - p\|^2$$

(Squared) Distance to Discrete Point Sets





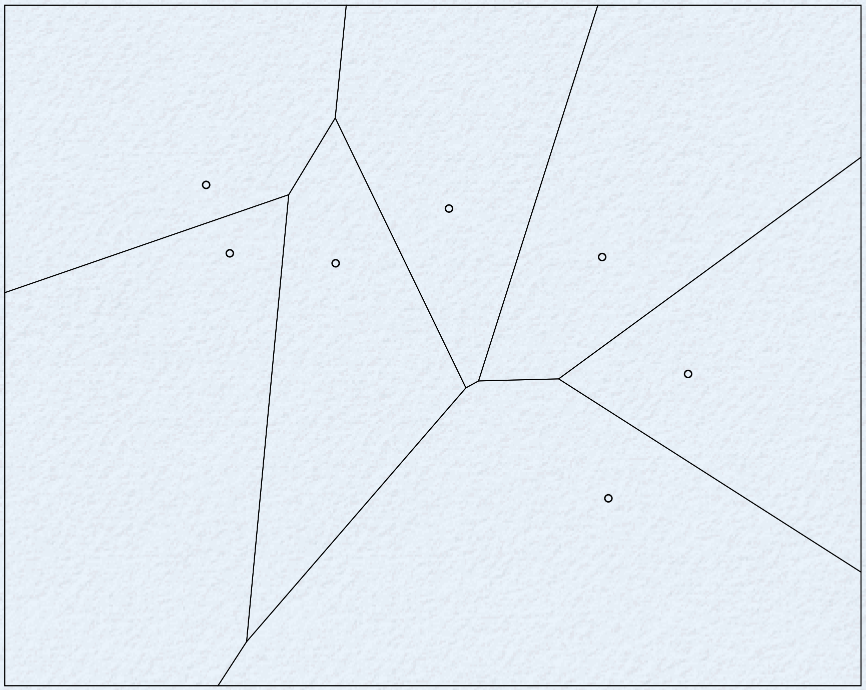
P is a discrete set of points

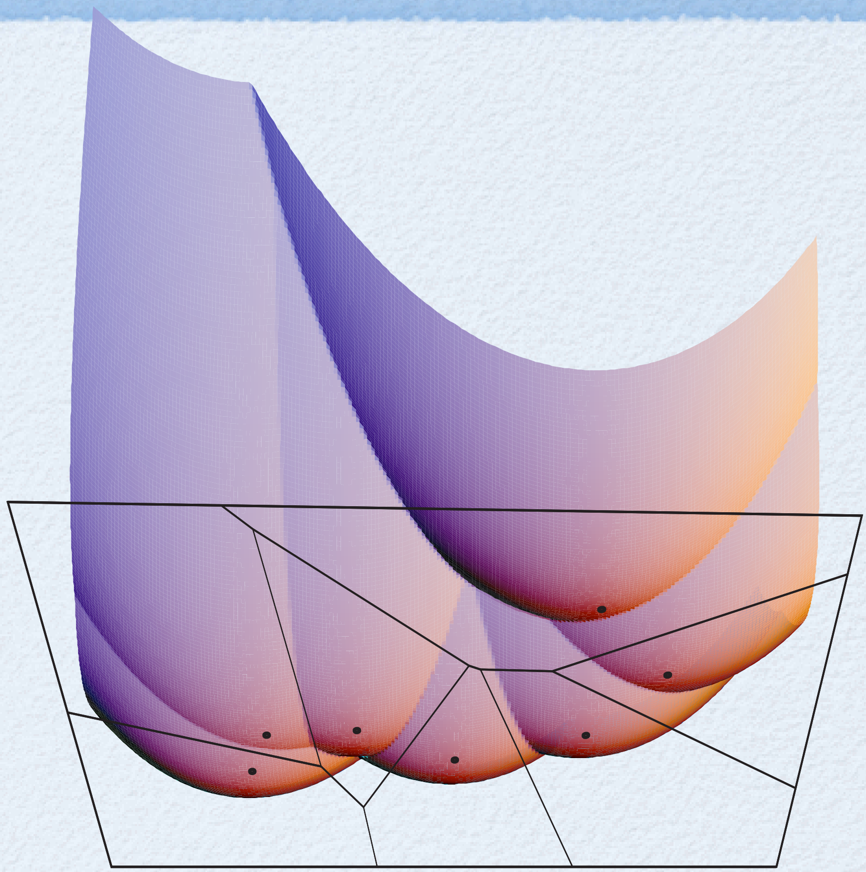
The squared distance function induced by P is

$$h(x) = \min_{p \in P} \|x - p\|^2$$

Observation. h is smooth at points with a unique closest point in P.

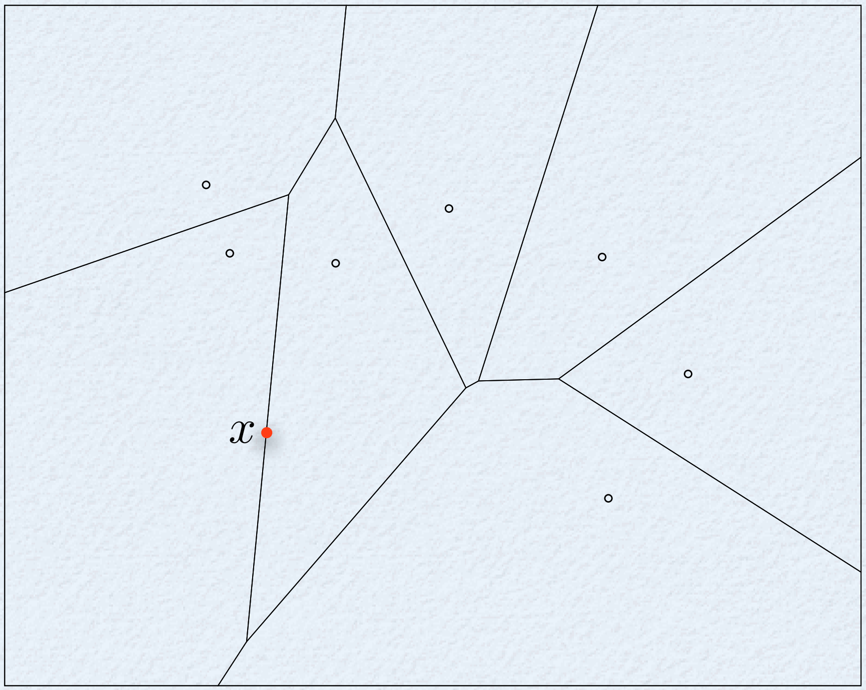
Generalized Gradient

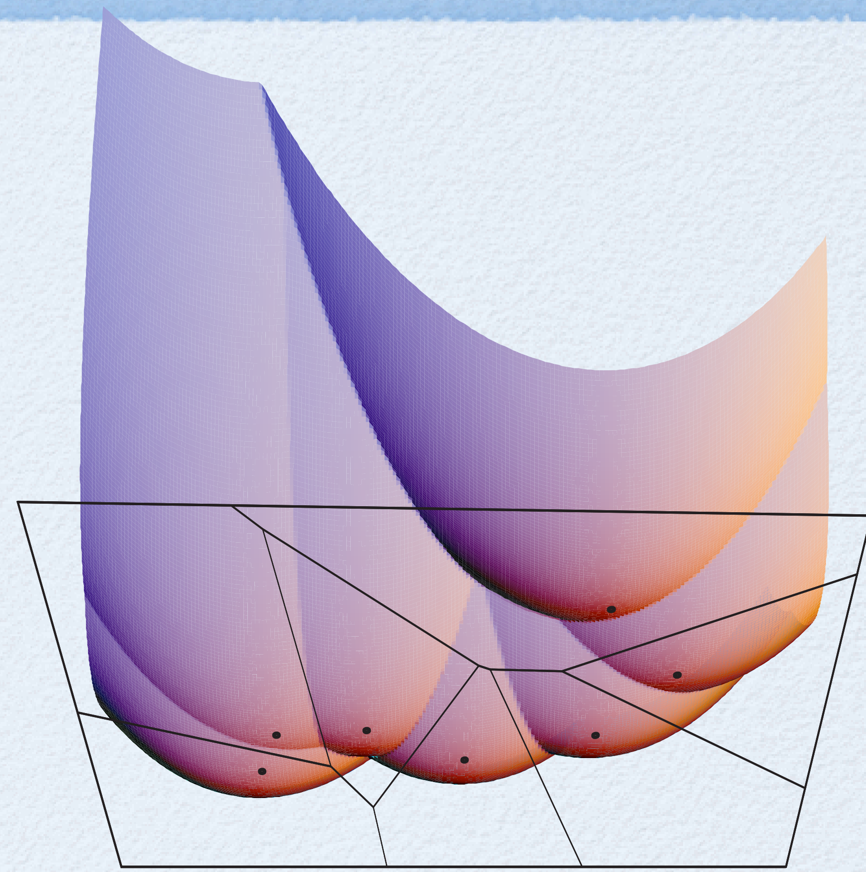




[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

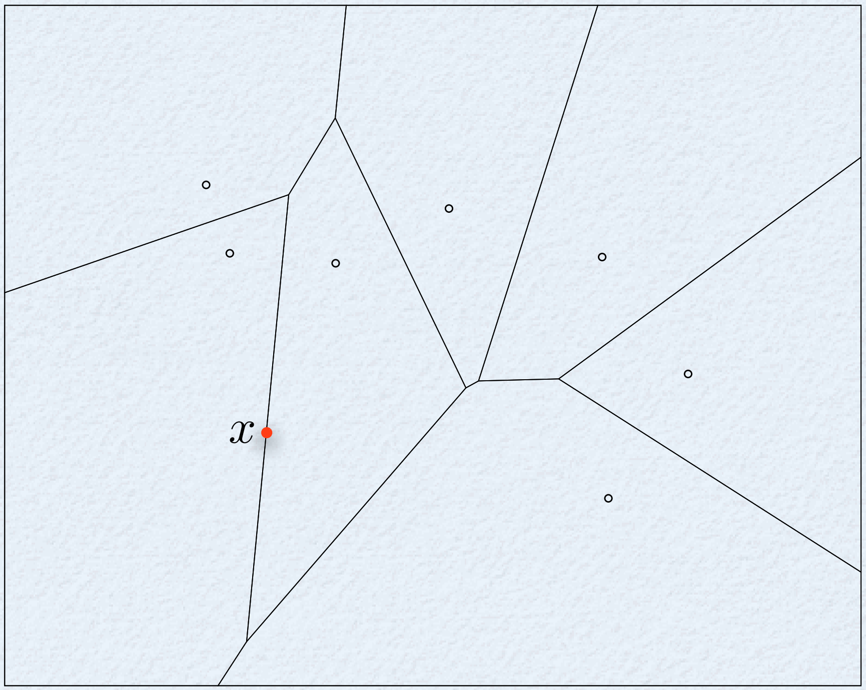
Generalized Gradient

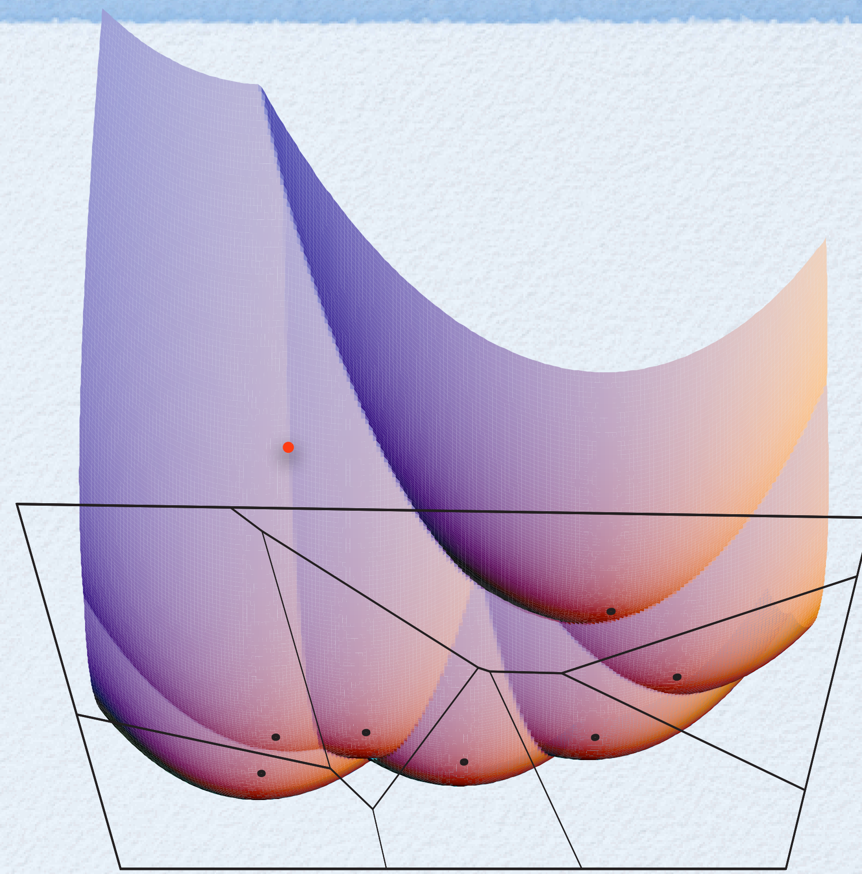




[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

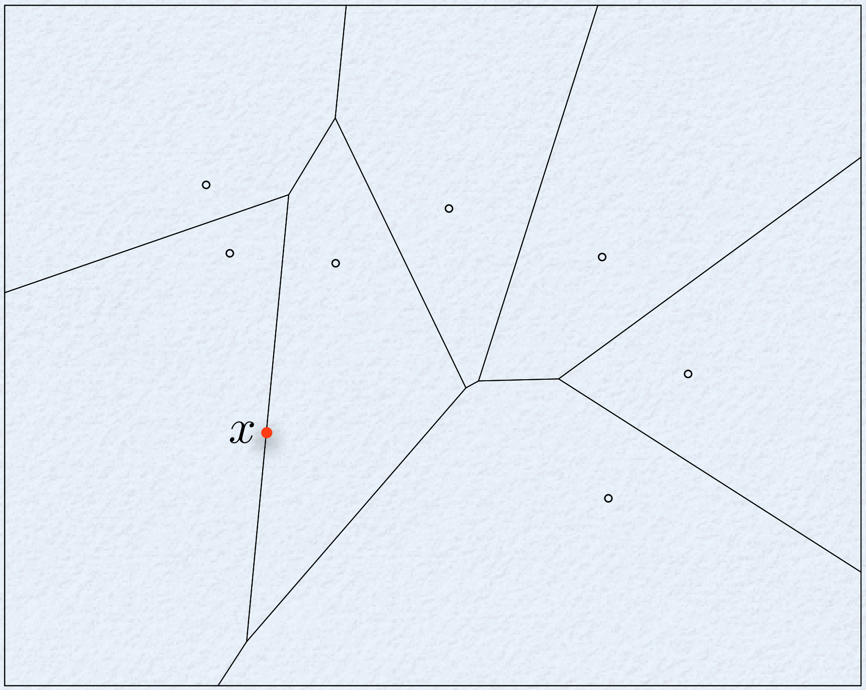
Generalized Gradient

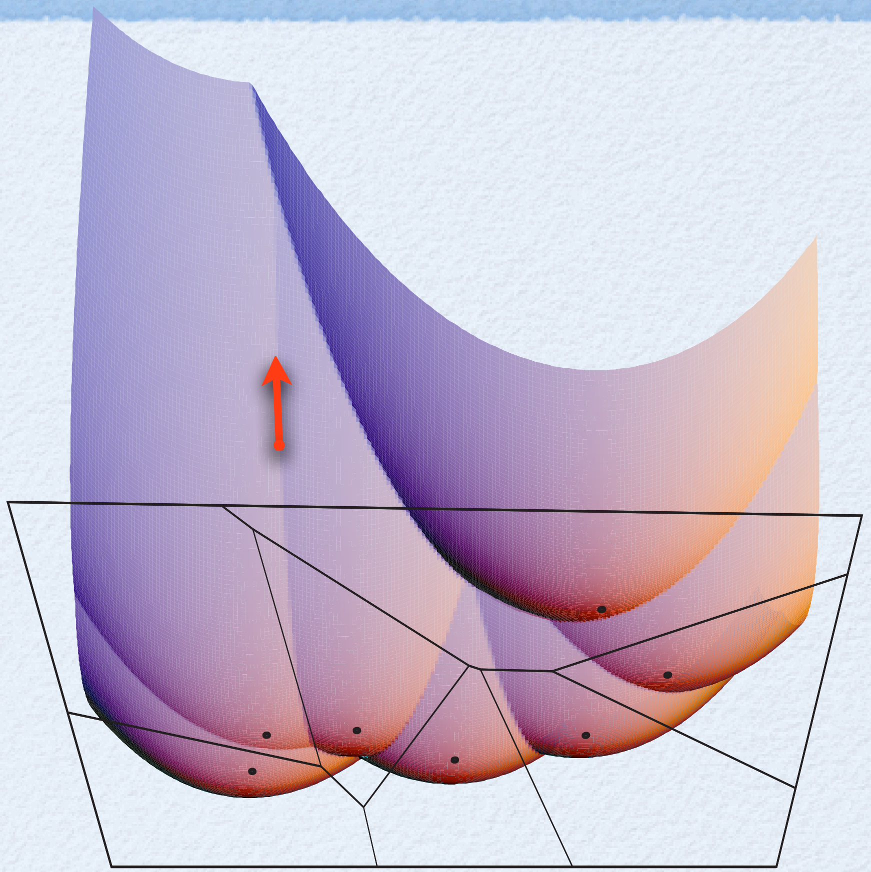




[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

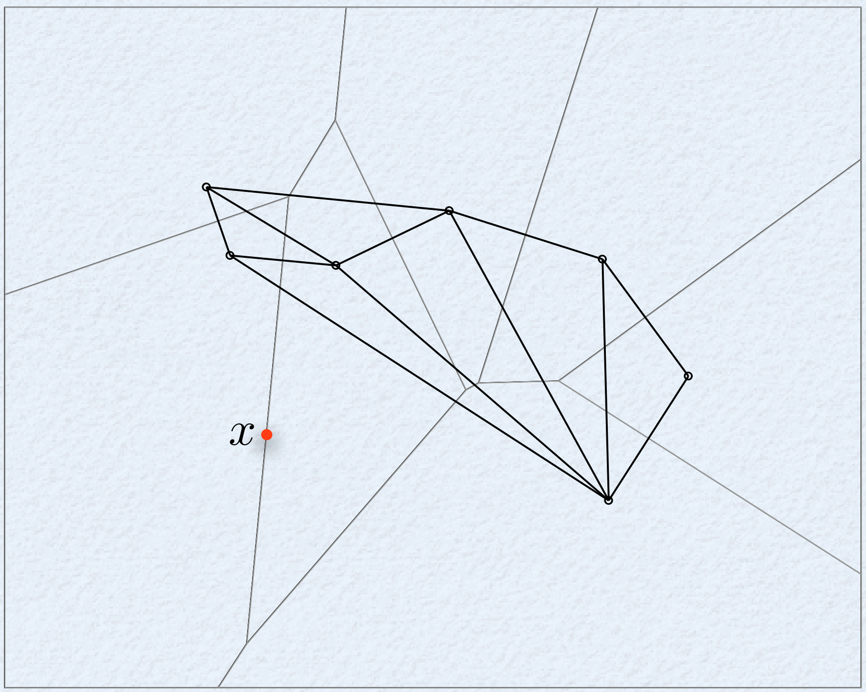
Generalized Gradient

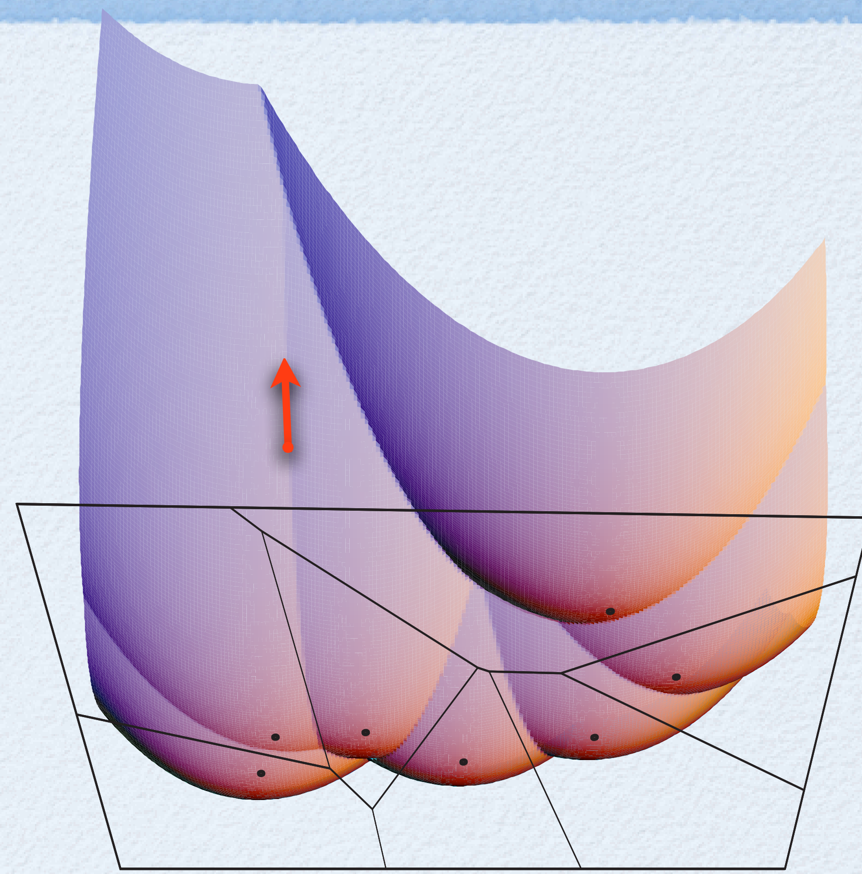




[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

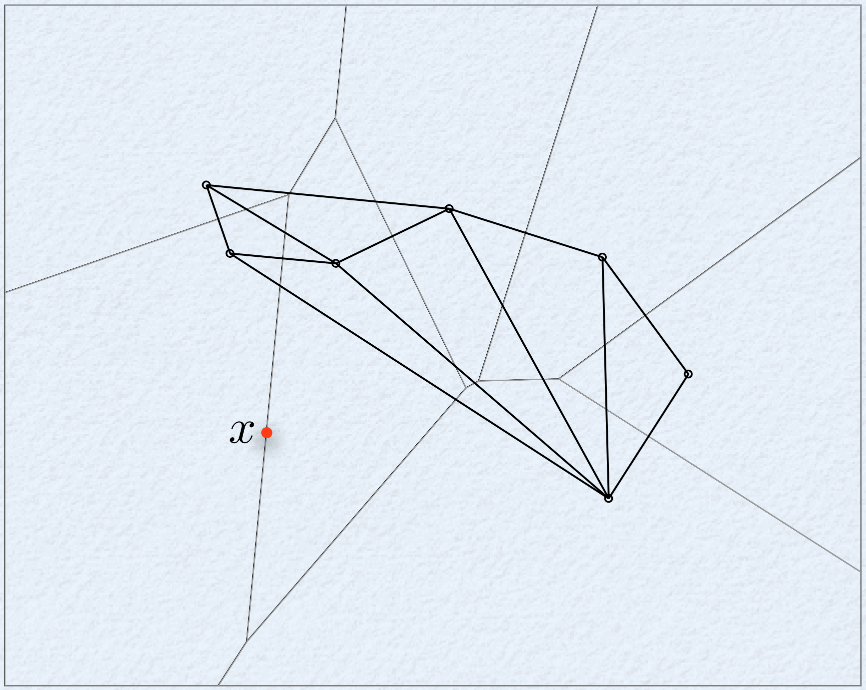
Generalized Gradient

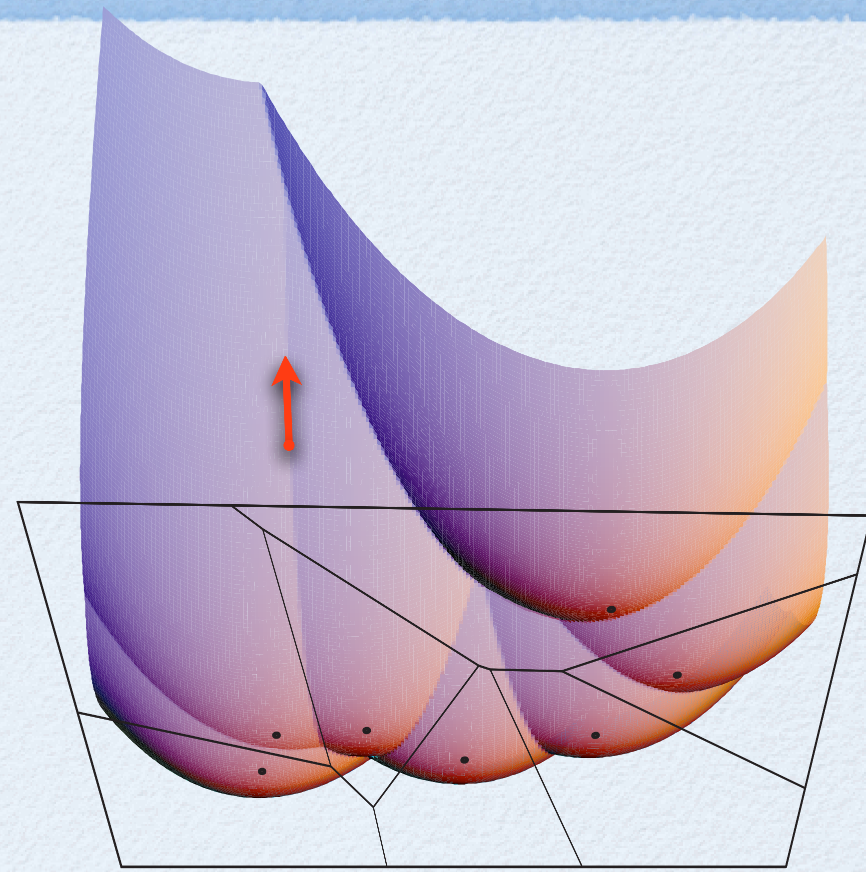




[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

Generalized Gradient

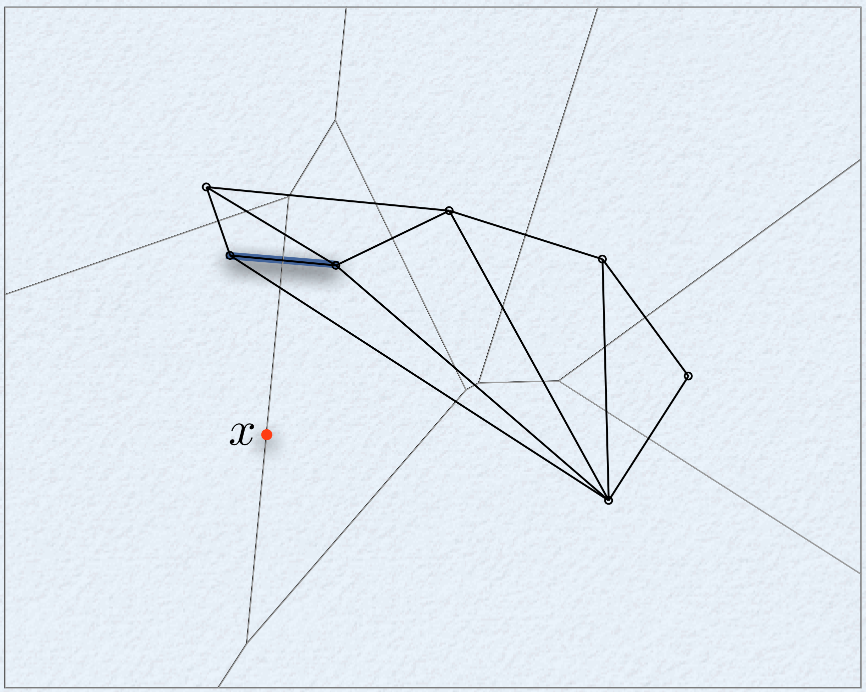


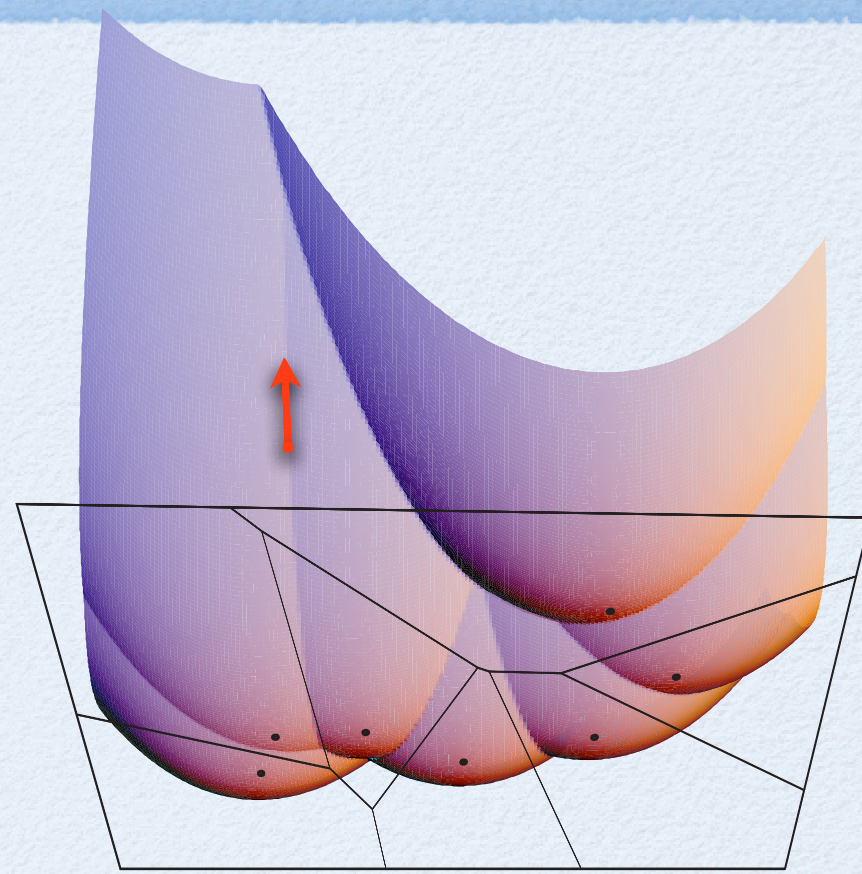


$V(x)$: lowest-dimensional Voronoi face containing x.

[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

Generalized Gradient



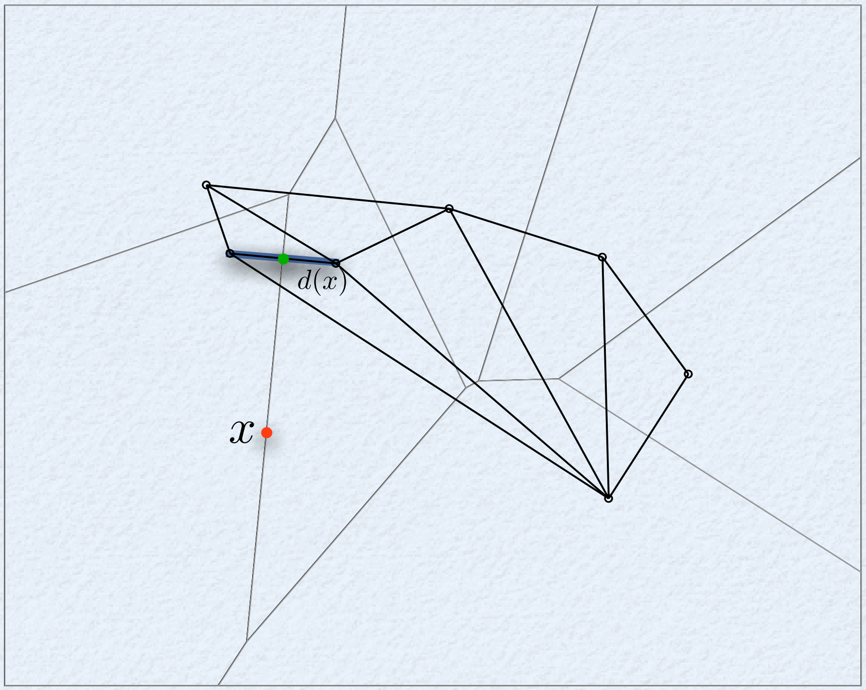


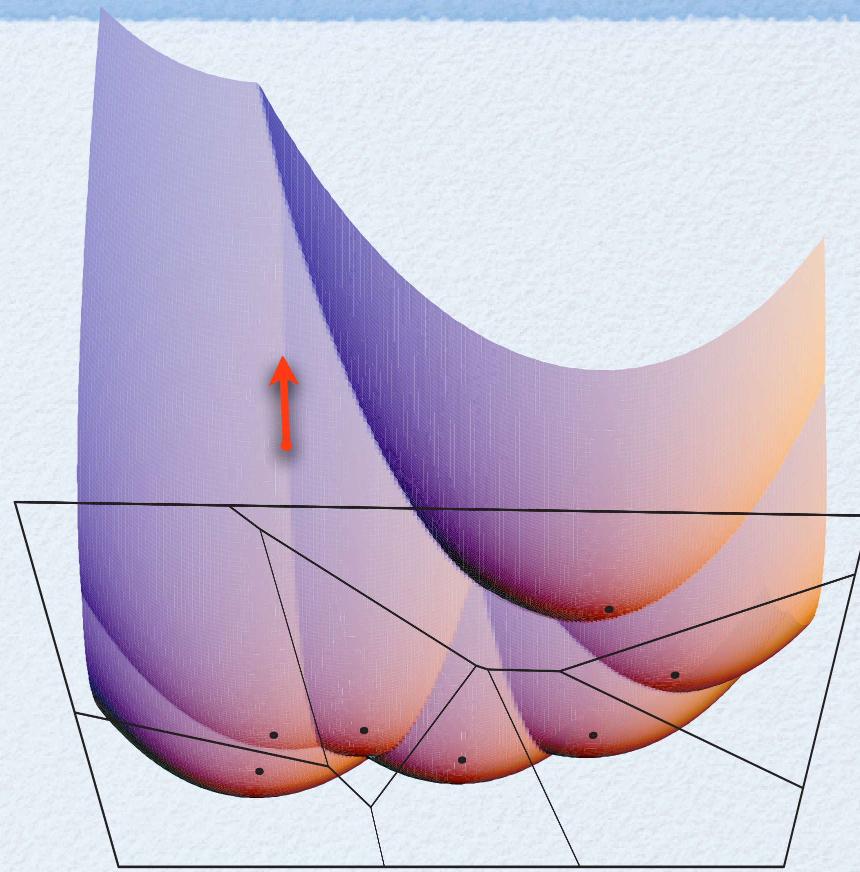
$V(x)$: lowest-dimensional Voronoi face containing x.

$D(x)$: Delaunay dual to $V(x)$.

[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

Generalized Gradient





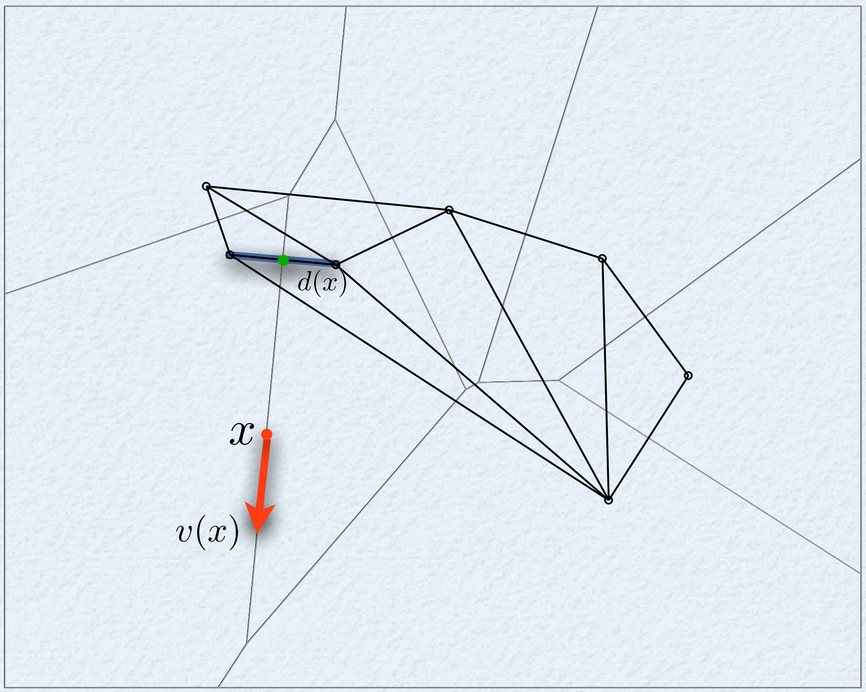
$V(x)$: lowest-dimensional Voronoi face containing x.

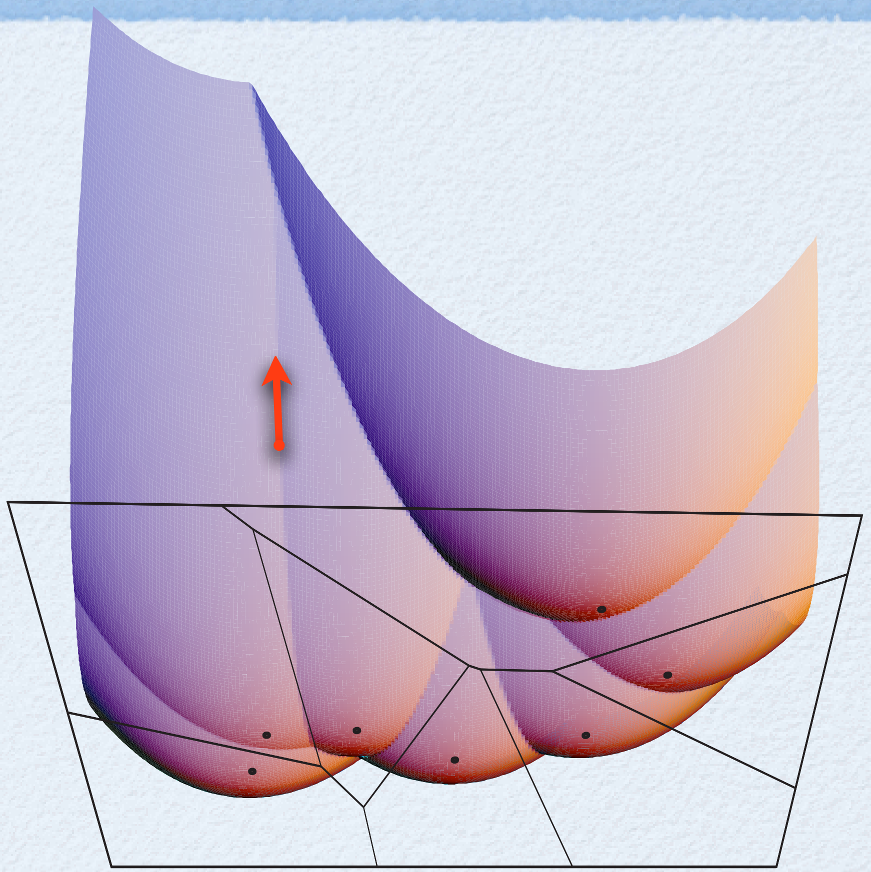
$D(x)$: Delaunay dual to $V(x)$.

The driver of x is the closest point to x in $D(x)$.

[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

Generalized Gradient





$V(x)$: lowest-dimensional Voronoi face containing x.

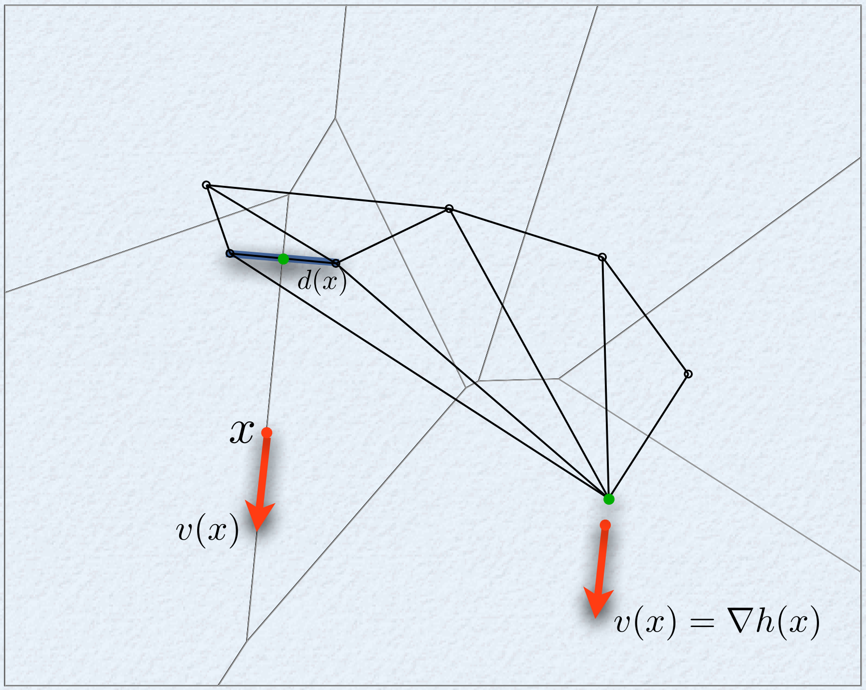
$D(x)$: Delaunay dual to $V(x)$.

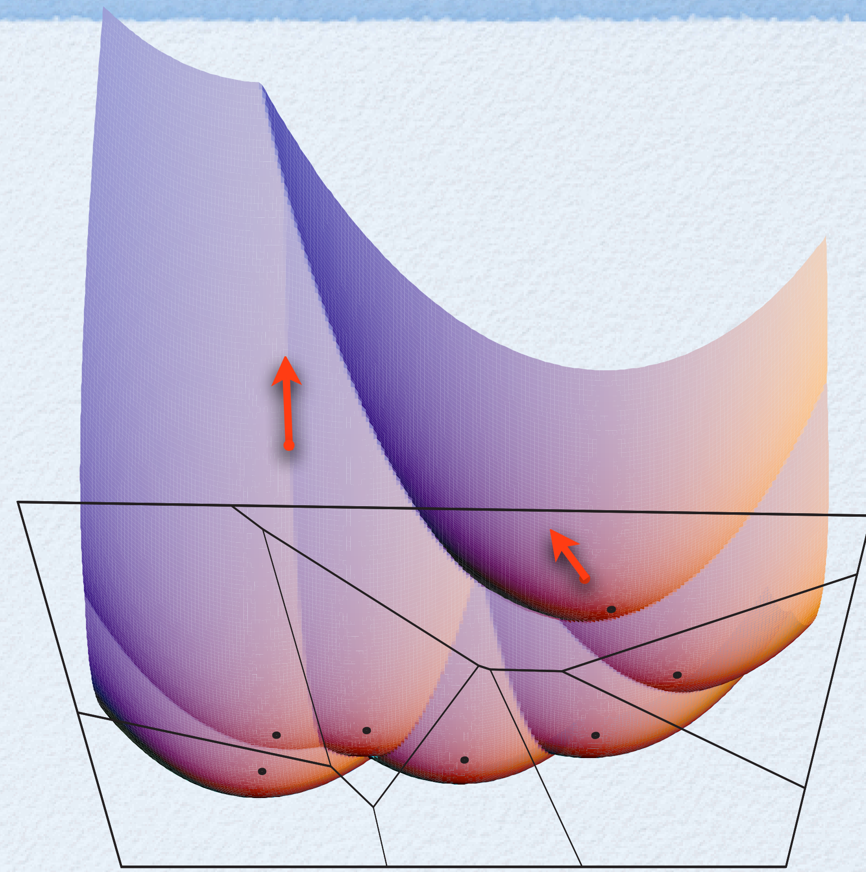
The driver of x is the closest point to x in $D(x)$.

$$v(x) = 2(x - d(x))$$

[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

Generalized Gradient





$V(x)$: lowest-dimensional Voronoi face containing x.

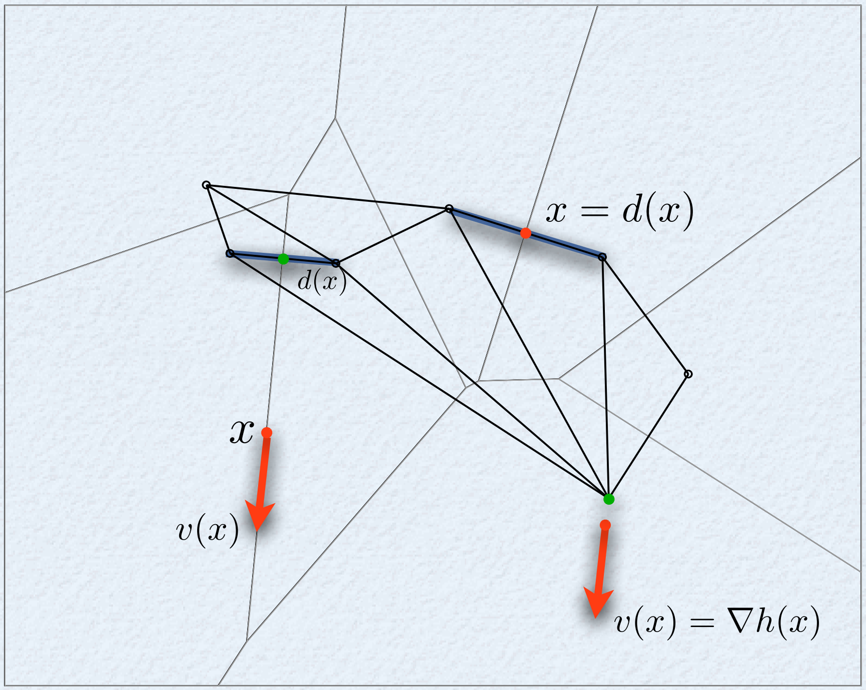
$D(x)$: Delaunay dual to $V(x)$.

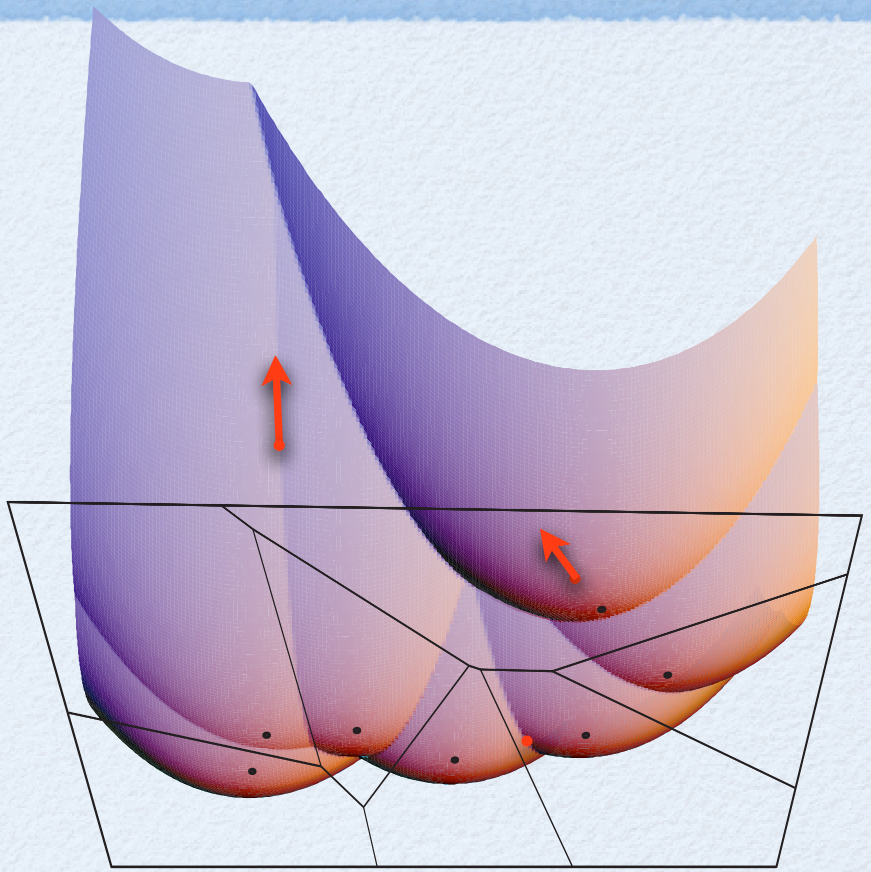
The driver of x is the closest point to x in $D(x)$.

$$v(x) = 2(x - d(x))$$

[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

Generalized Gradient





$V(x)$: lowest-dimensional Voronoi face containing x.

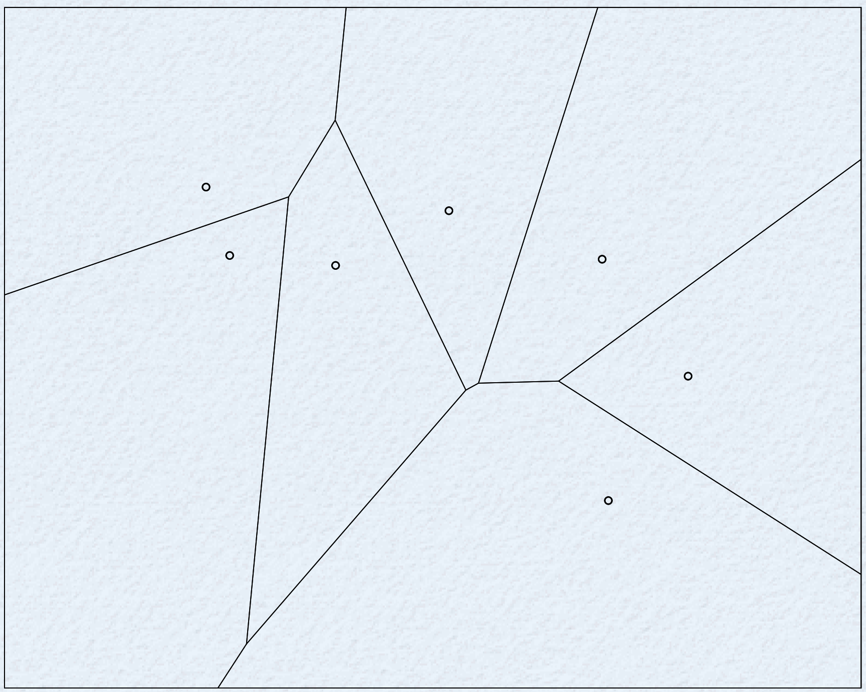
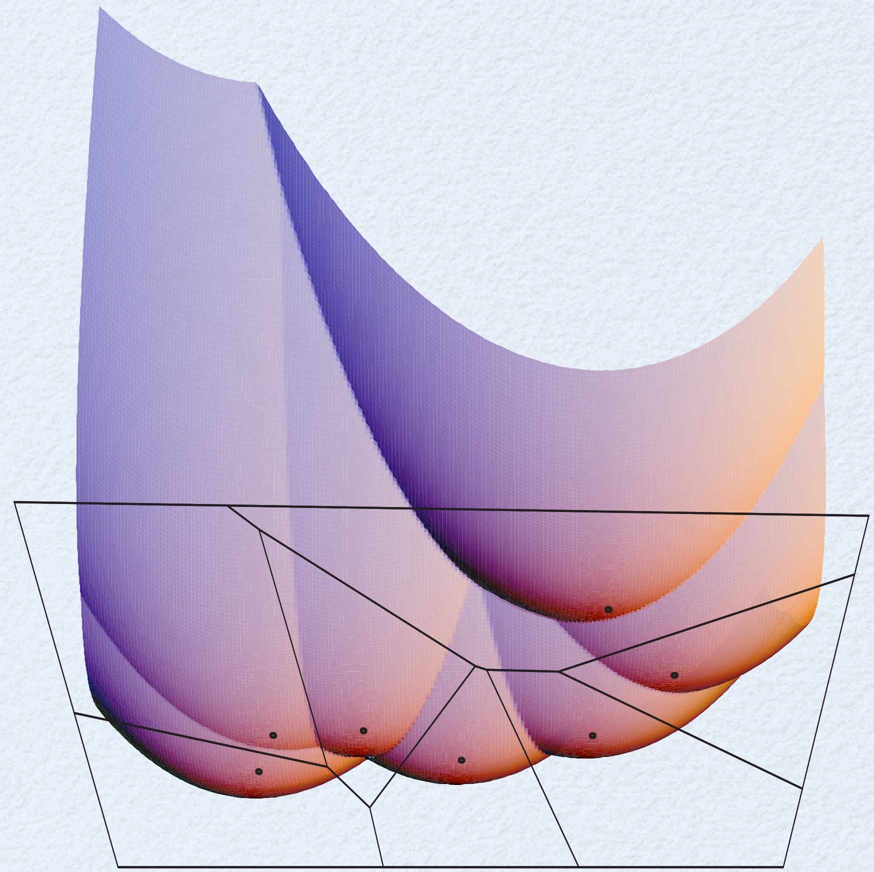
$D(x)$: Delaunay dual to $V(x)$.

The driver of x is the closest point to x in $D(x)$.

$$v(x) = 2(x - d(x))$$

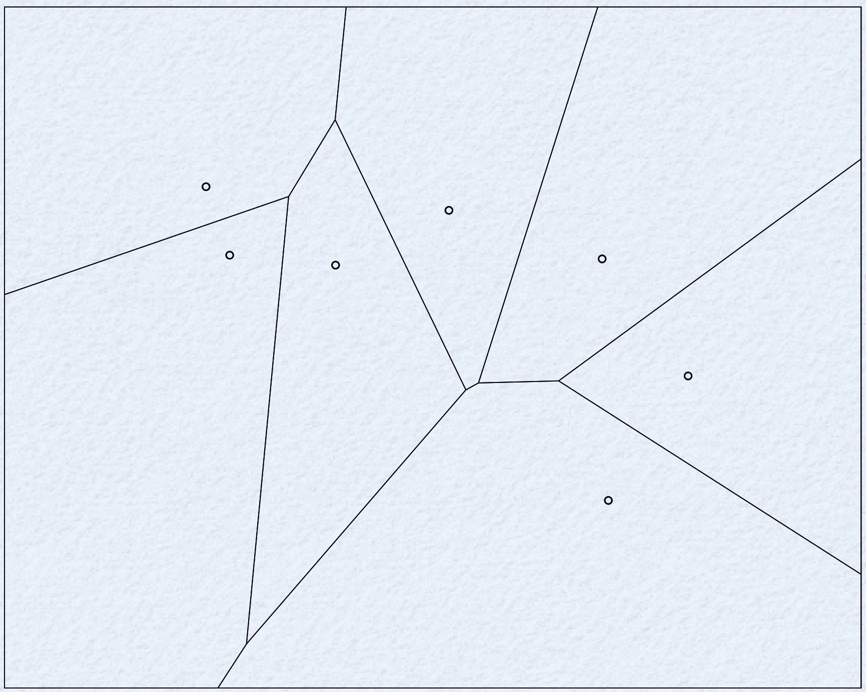
[Giesen-John'03] J. Giesen, M. John, Flow complex: A data structure for geometric modeling. SODA, 2003.

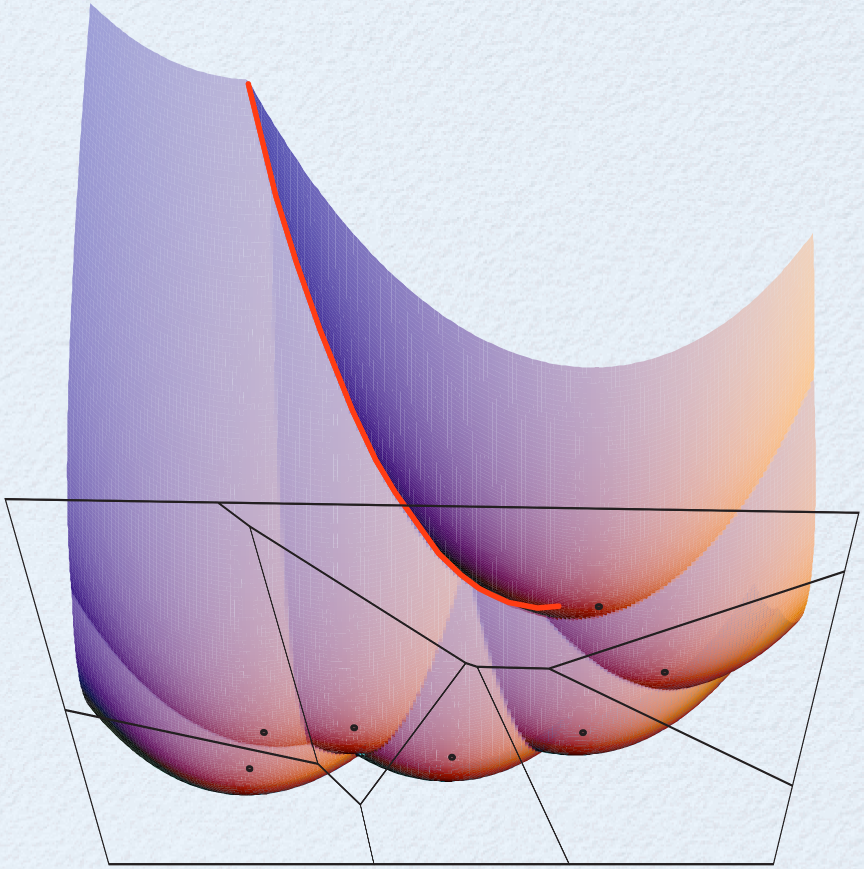
Integrating v

Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

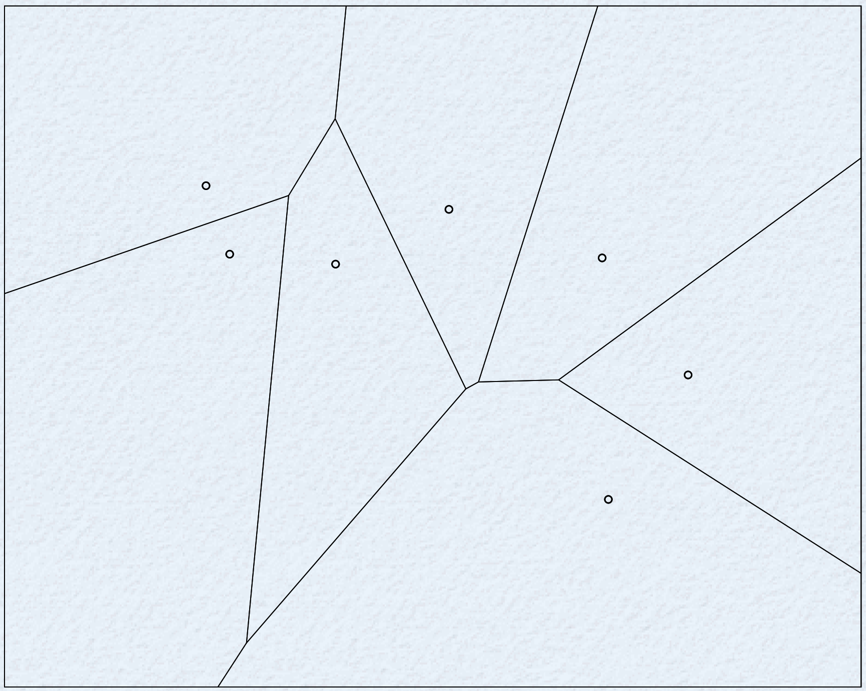
Integrating v

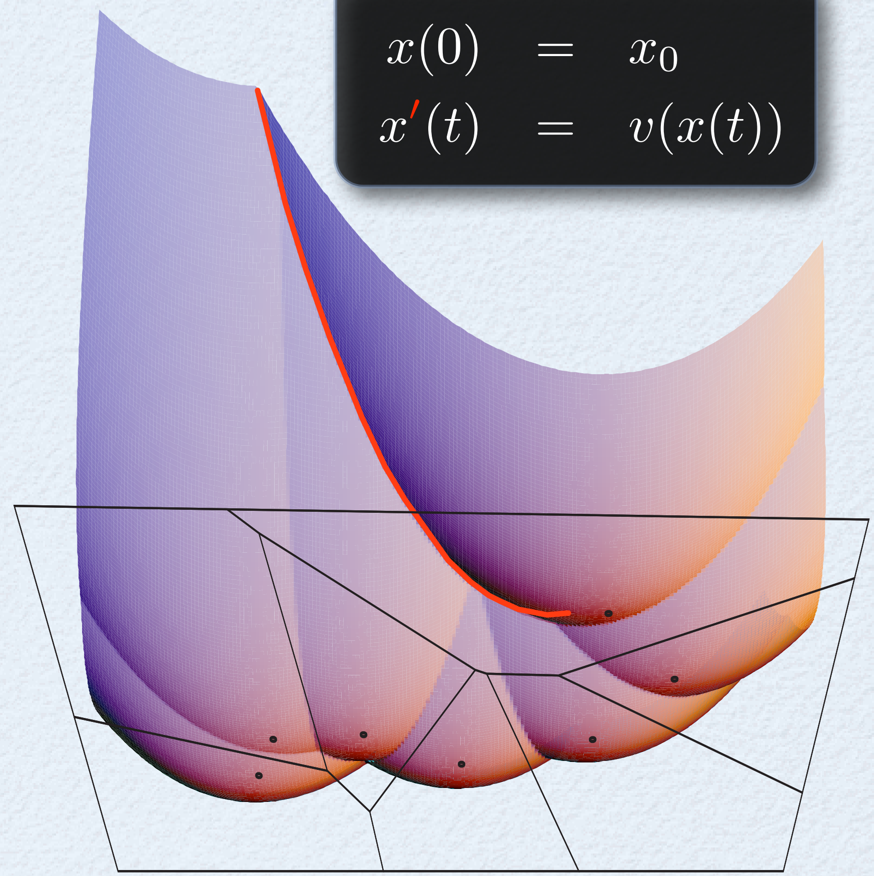




Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

Integrating v

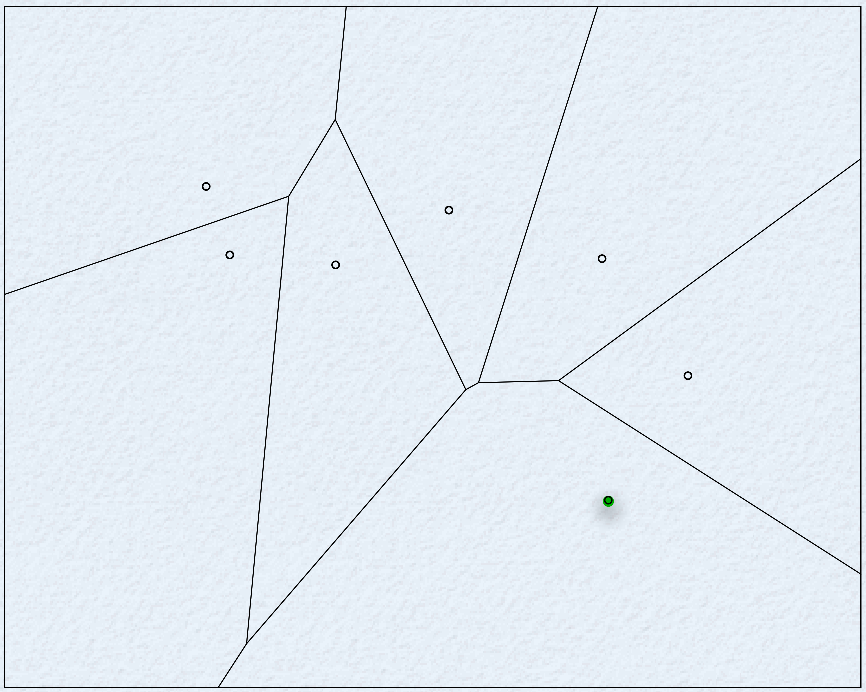


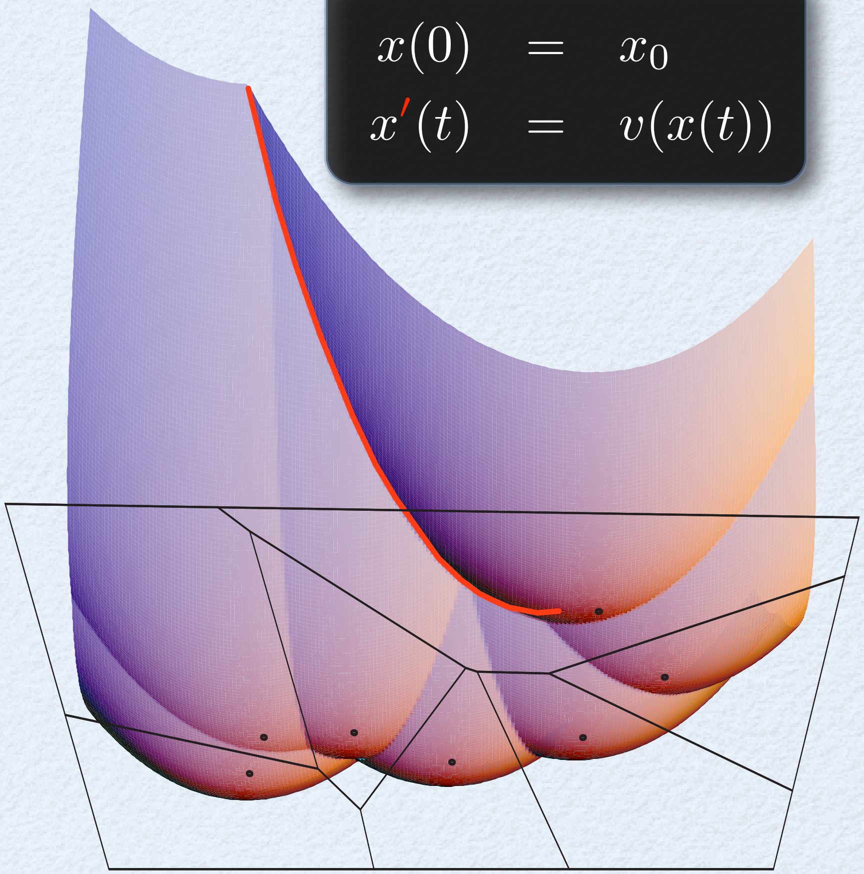


$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$

Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

Integrating v

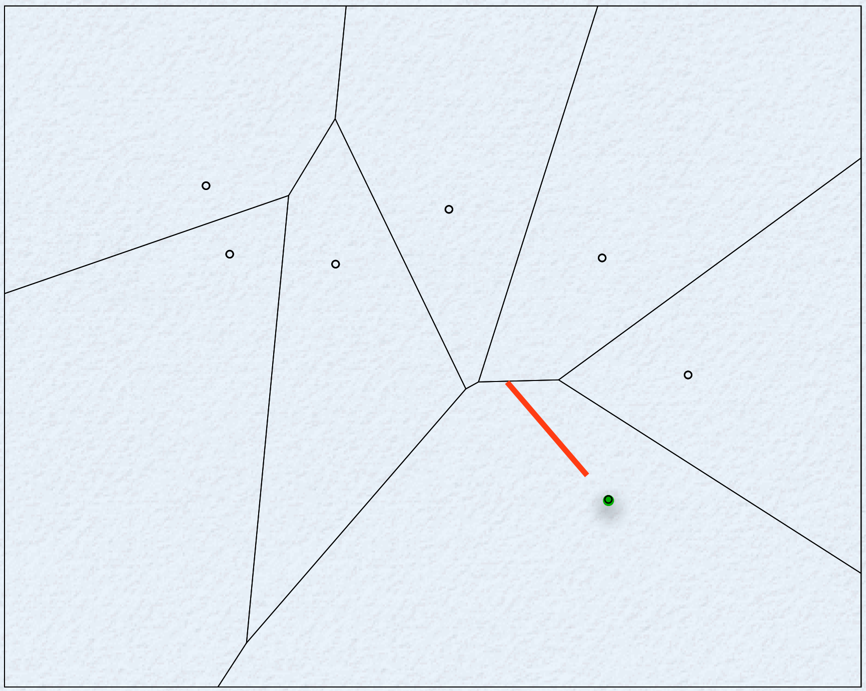




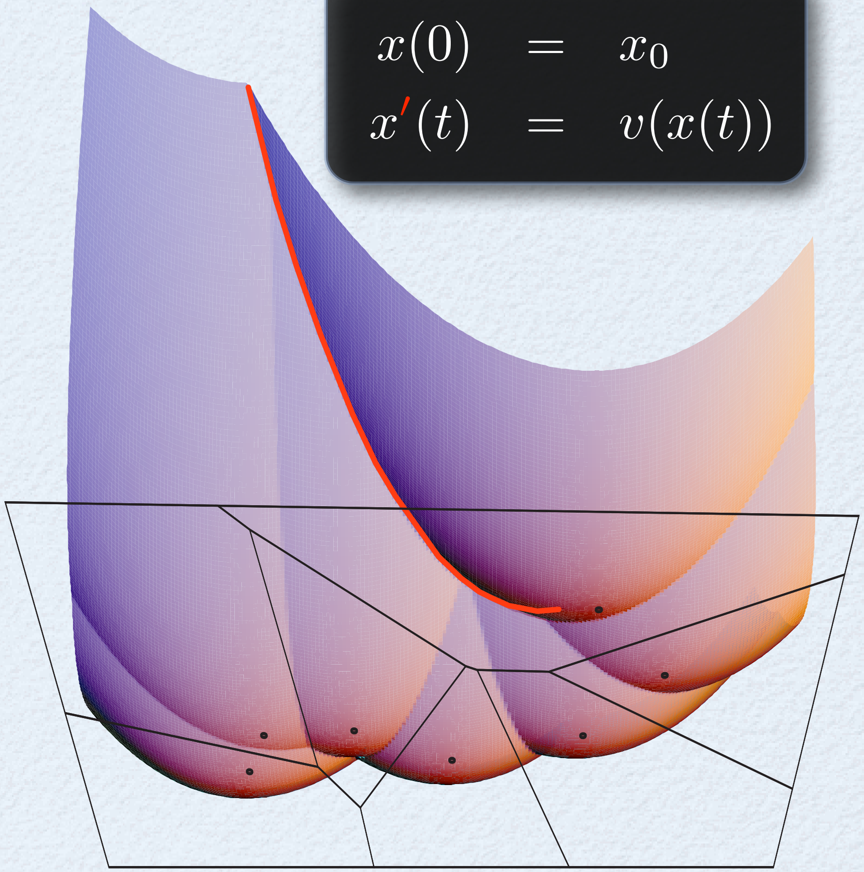
$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$

Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

Integrating v

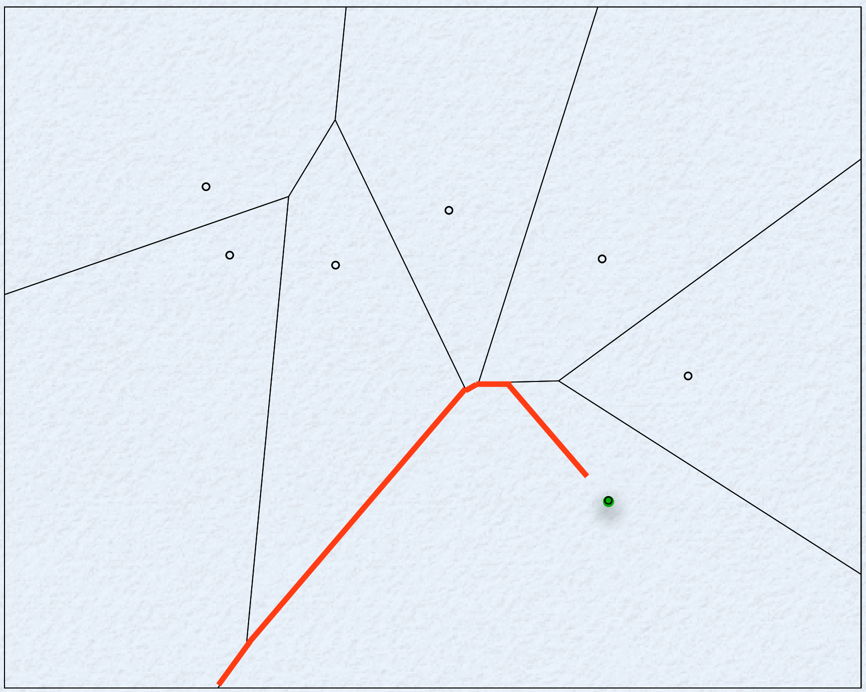


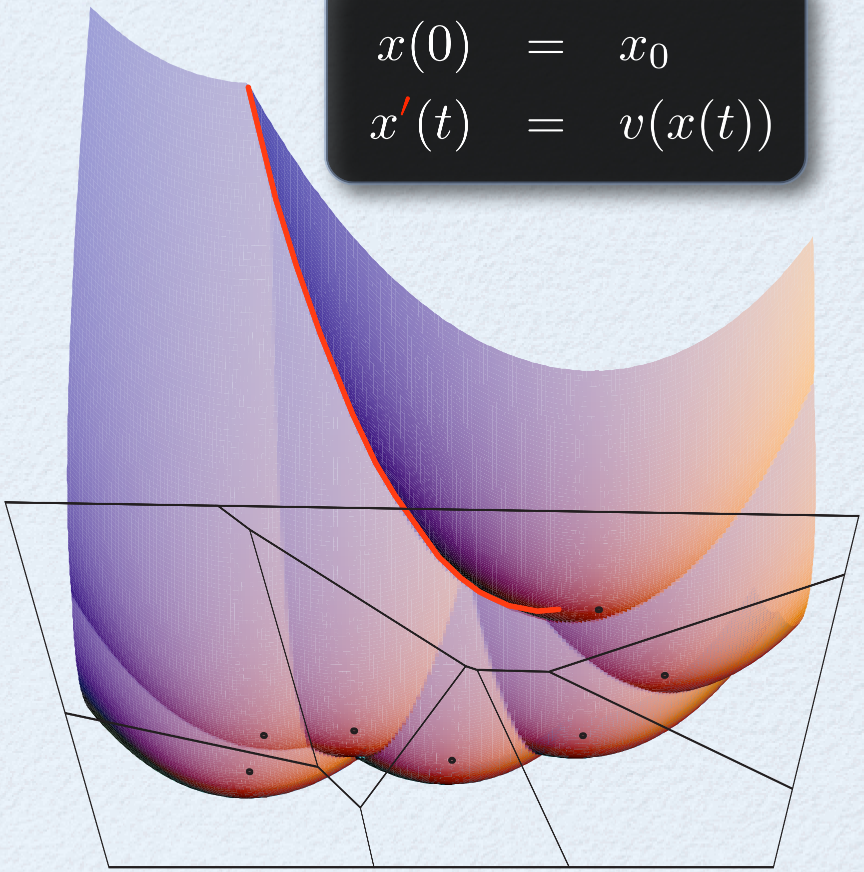
$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$



Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

Integrating v



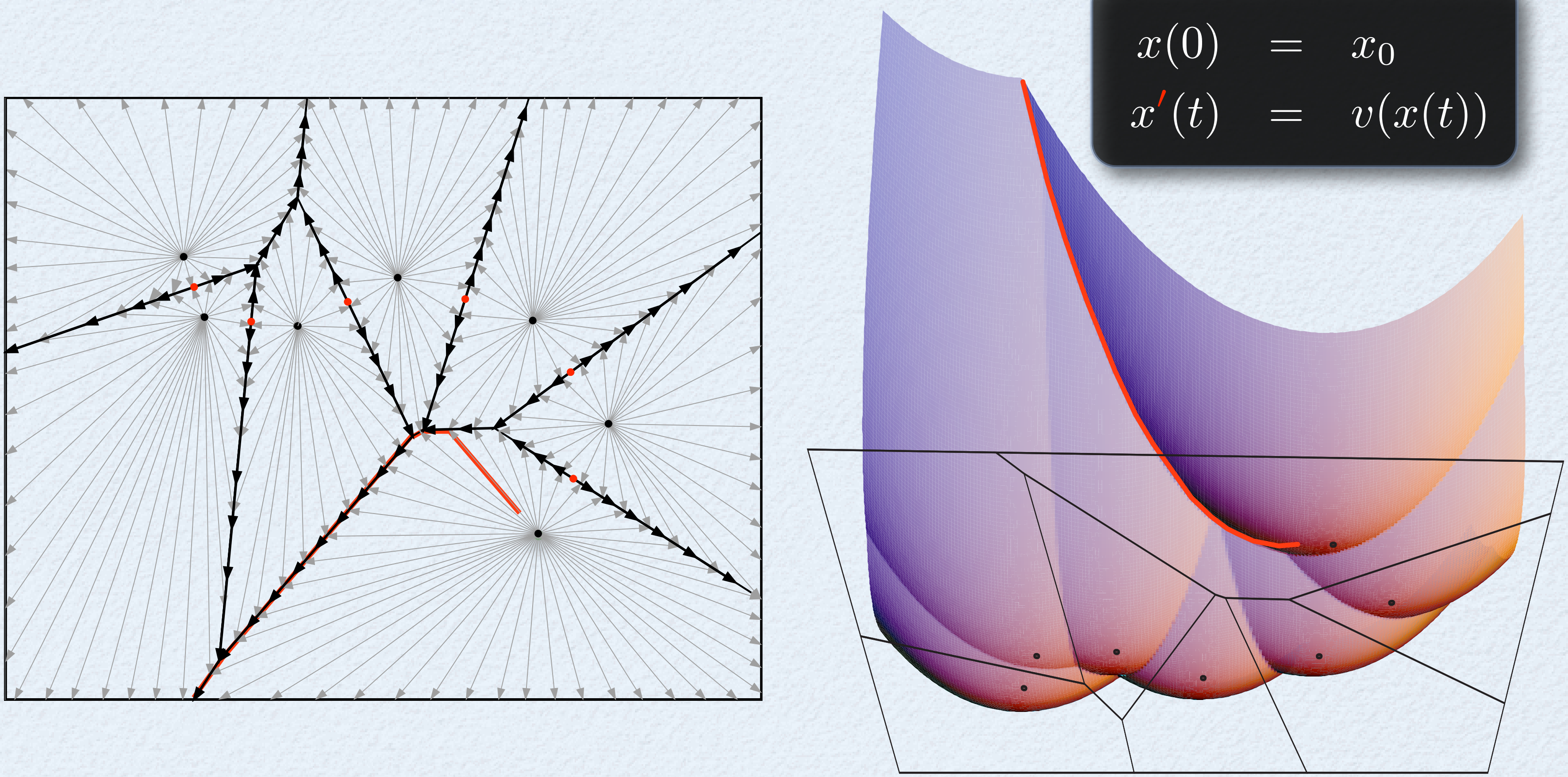


$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$

Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

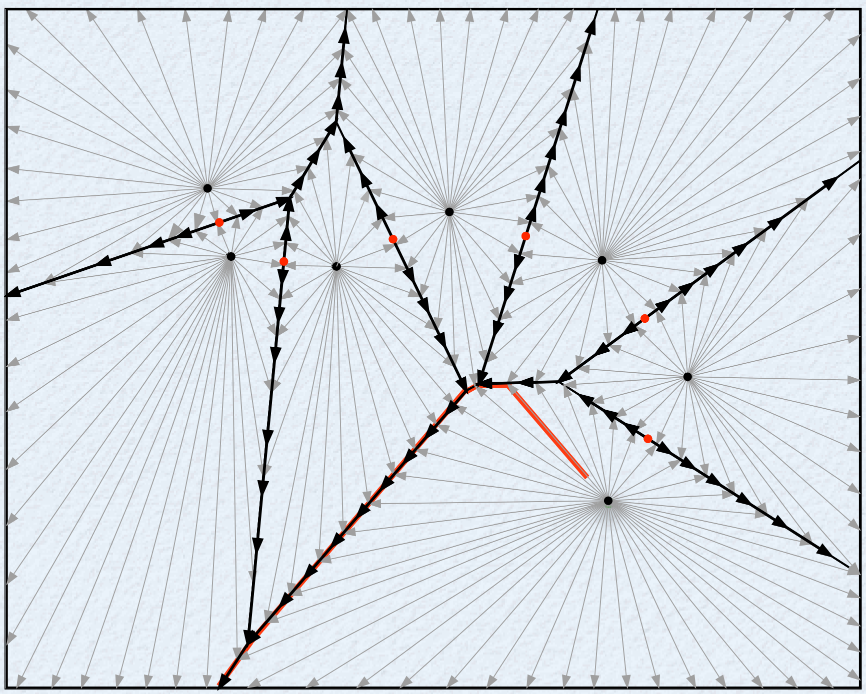
Integrating v

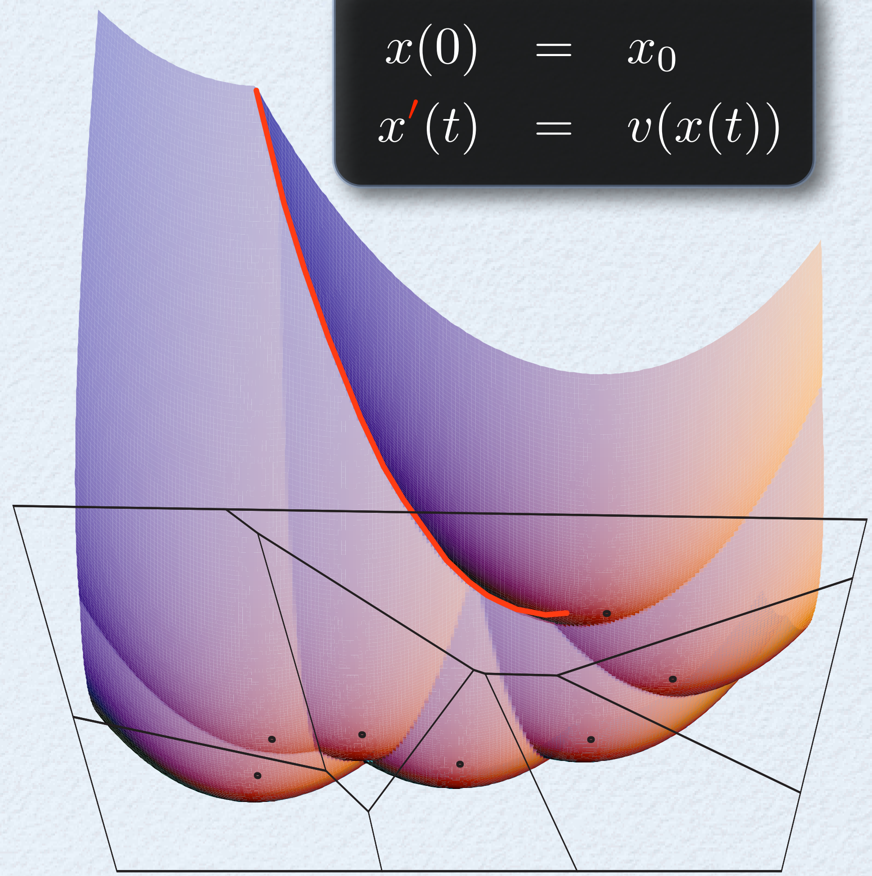
$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$



Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

Integrating v



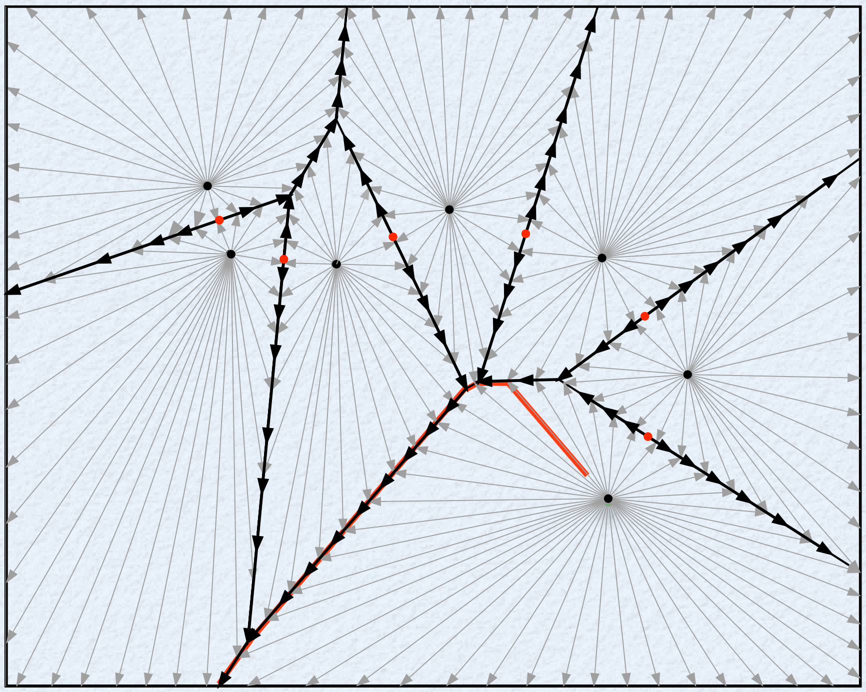


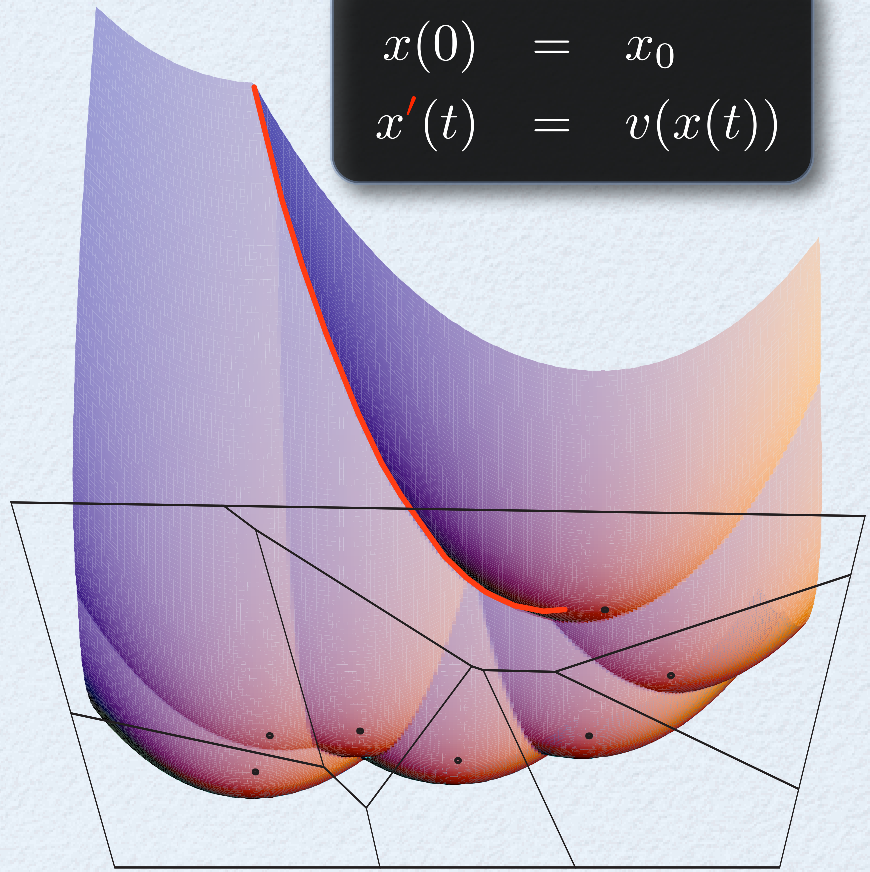
$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$

Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

$\phi(t, x) = y$ means "starting at x and going for time t we reach y".

Integrating v





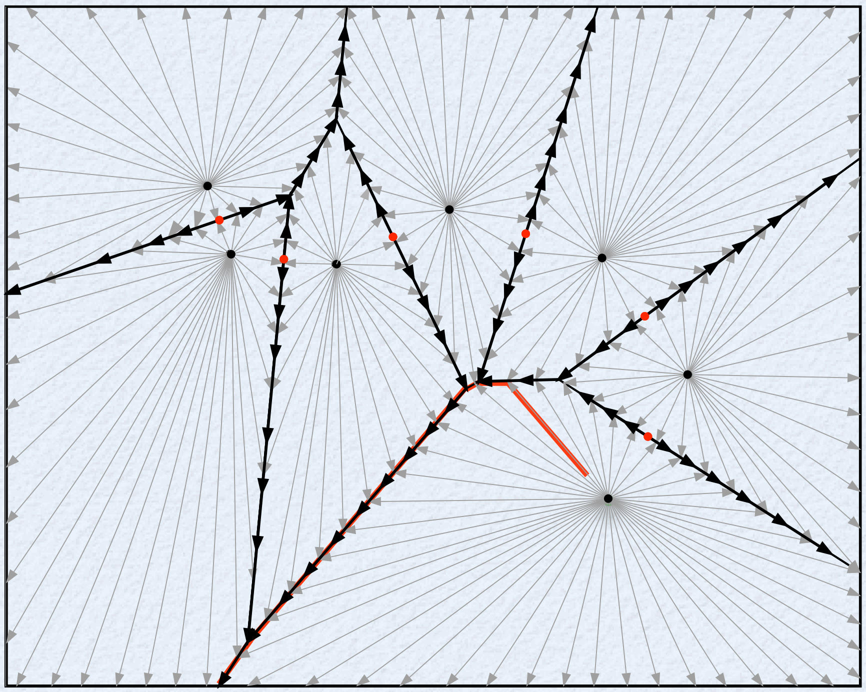
$$\begin{aligned} x(0) &= x_0 \\ x'(t) &= v(x(t)) \end{aligned}$$

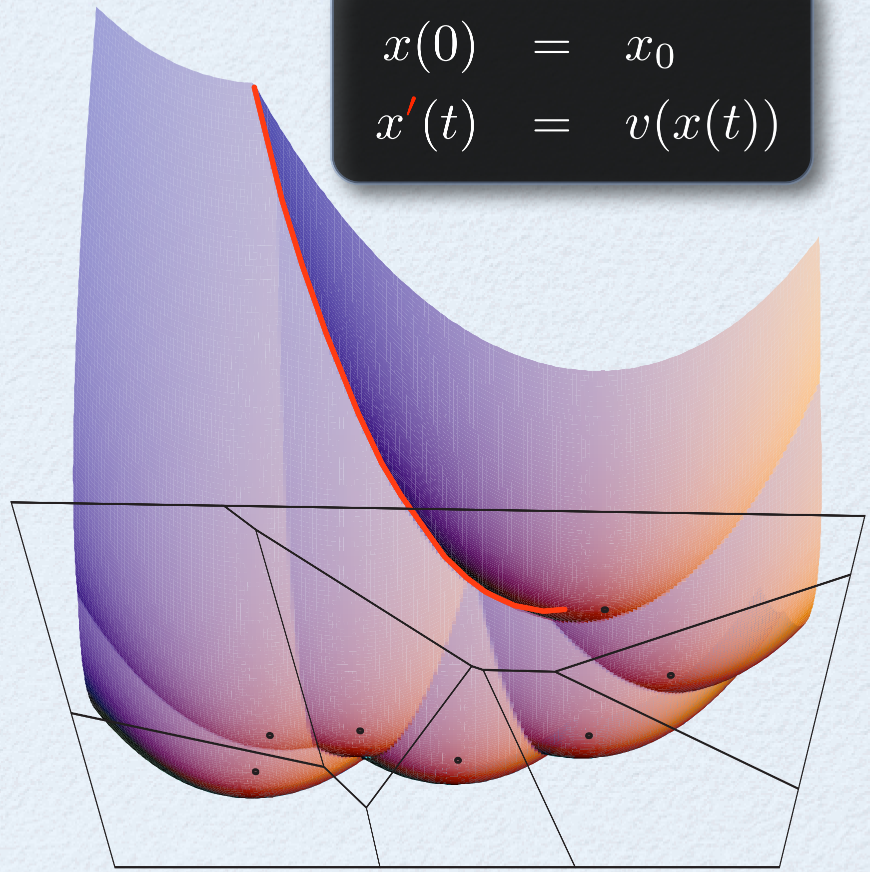
Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

$\phi(t, x) = y$ means "starting at x and going for time t we reach y".

$$\phi(x) = \{\phi(t, x) : t \geq 0\}$$

Integrating v





$$
\begin{aligned}
x(0) &= x_0 \\
x'(t) &= v(x(t))
\end{aligned}
$$

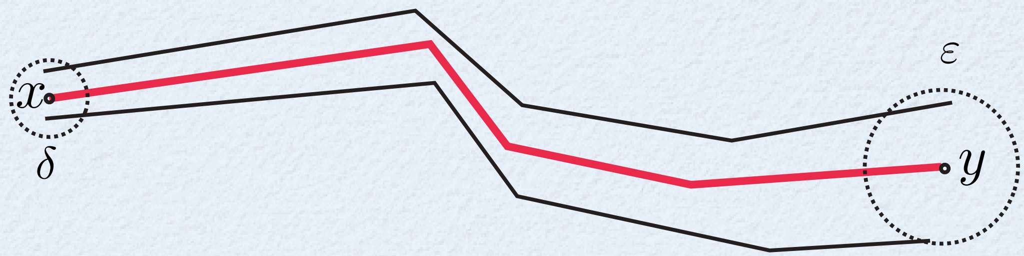
Moving at point x in with speed $v(x)$ results a flow map $\phi : \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}^n$.

$\phi(t, x) = y$ means "starting at x and going for time t we reach y".

$$\phi(x) = \{\phi(t, x) : t \geq 0\} \qquad \phi(X) = \bigcup_{x \in X} \phi(x)$$

Continuity of the Induced Flow

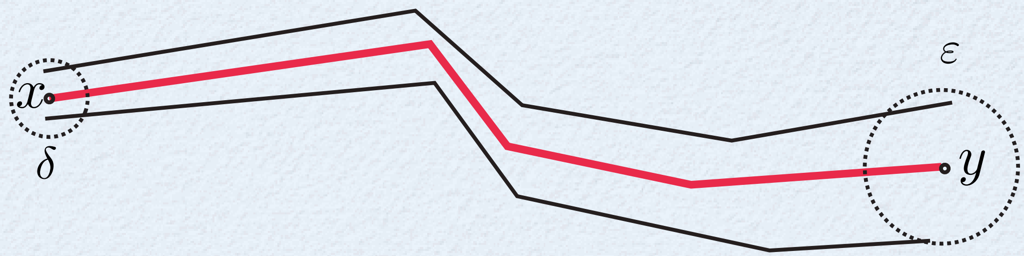
Theorem. The flow map $\phi : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ is continuous on both variables.



$y = \phi(t, x)$

Continuity of the Induced Flow

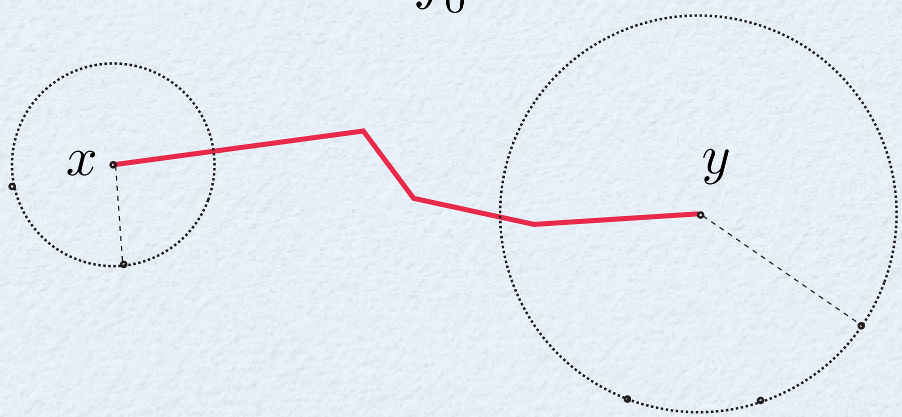
Theorem. The flow map $\phi : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n$ is continuous on both variables.



$y = \phi(t, x)$

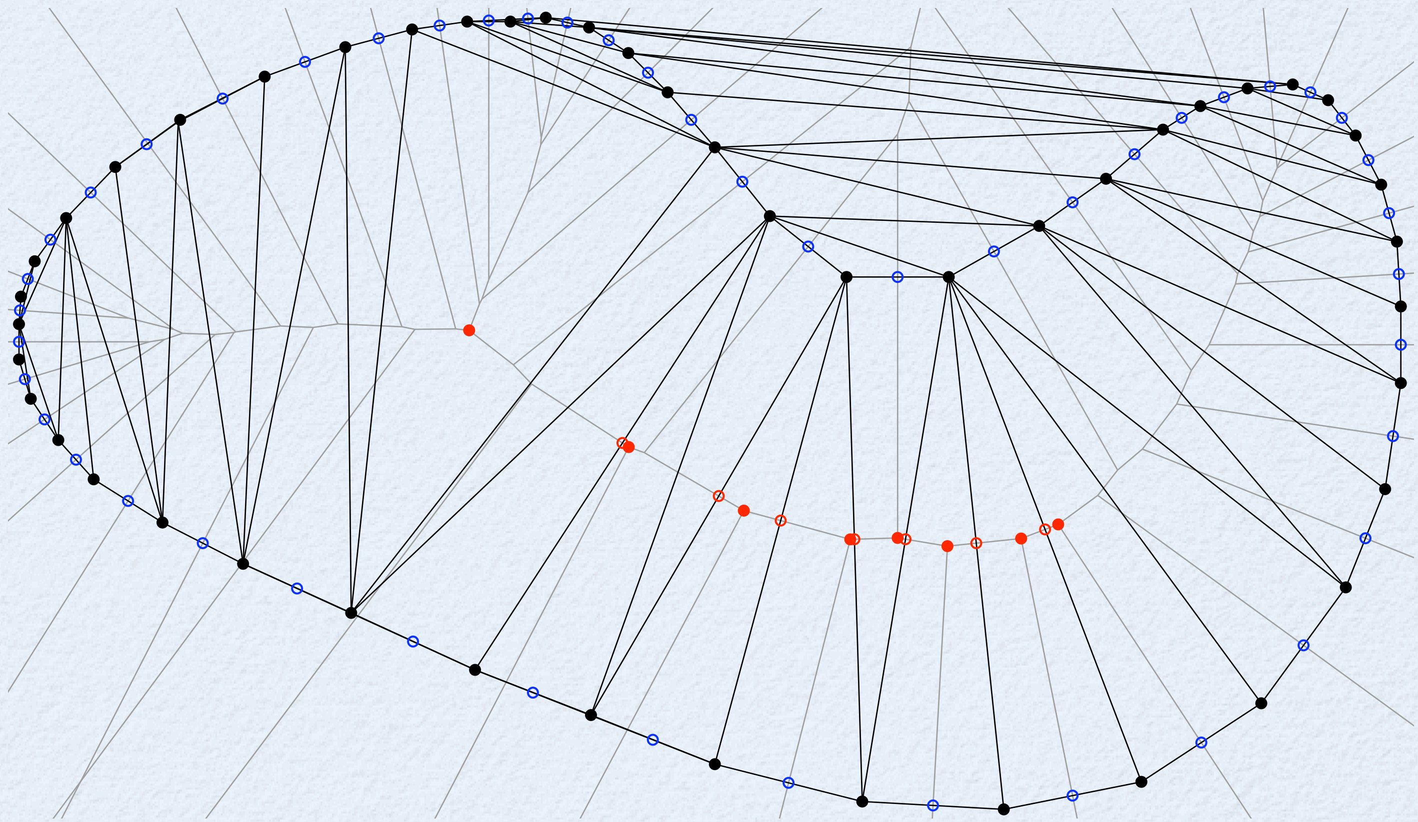
Theorem. For $y = \phi(t, x)$,

$$h(y) = h(x) + \int_0^t \|v(\phi(\tau, x))\|^2 d\tau.$$



Critical Points of Distance Function

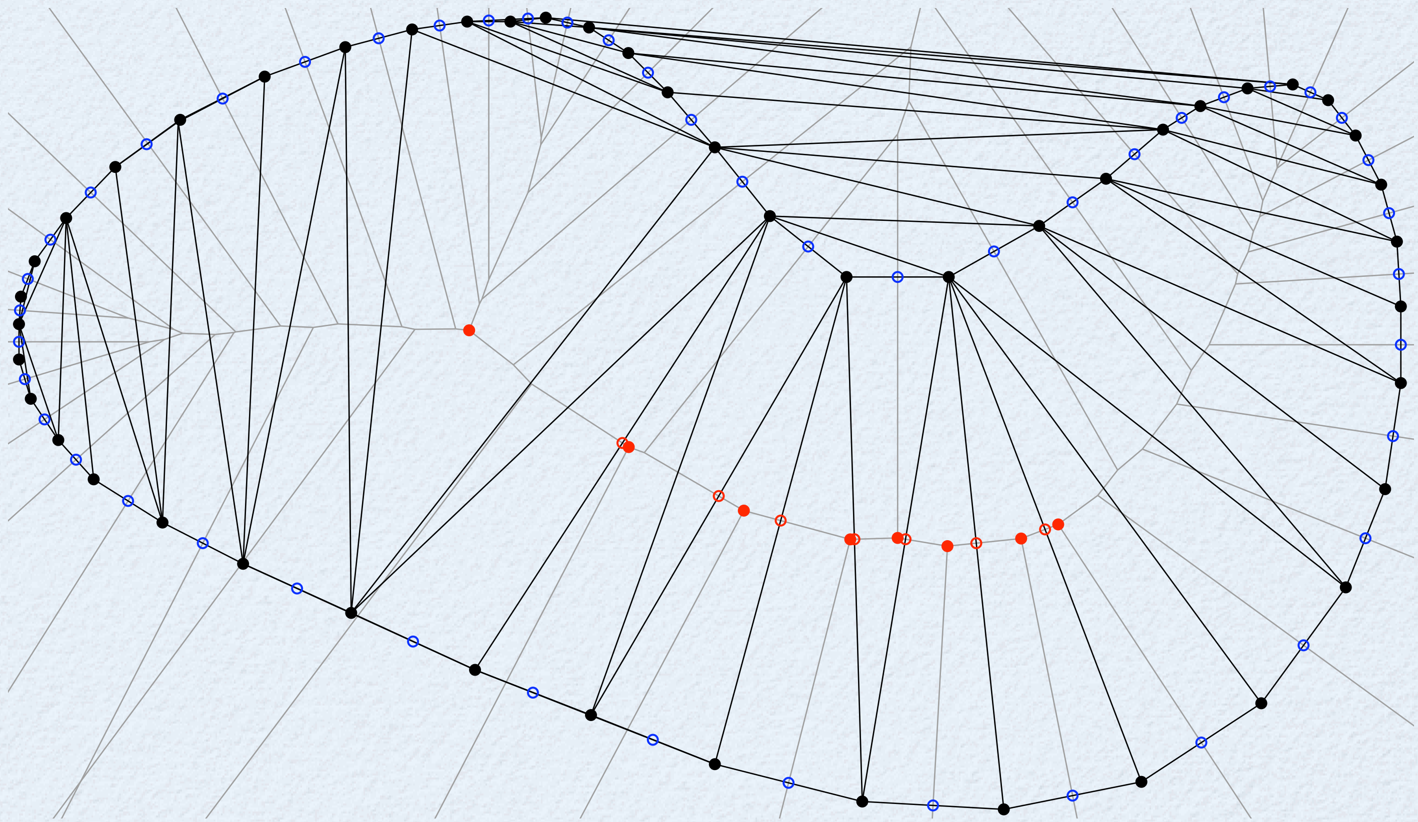
A point c with $v(c) = 0$ is called critical.



Critical Points of Distance Function

A point c with $v(c) = 0$ is called critical.

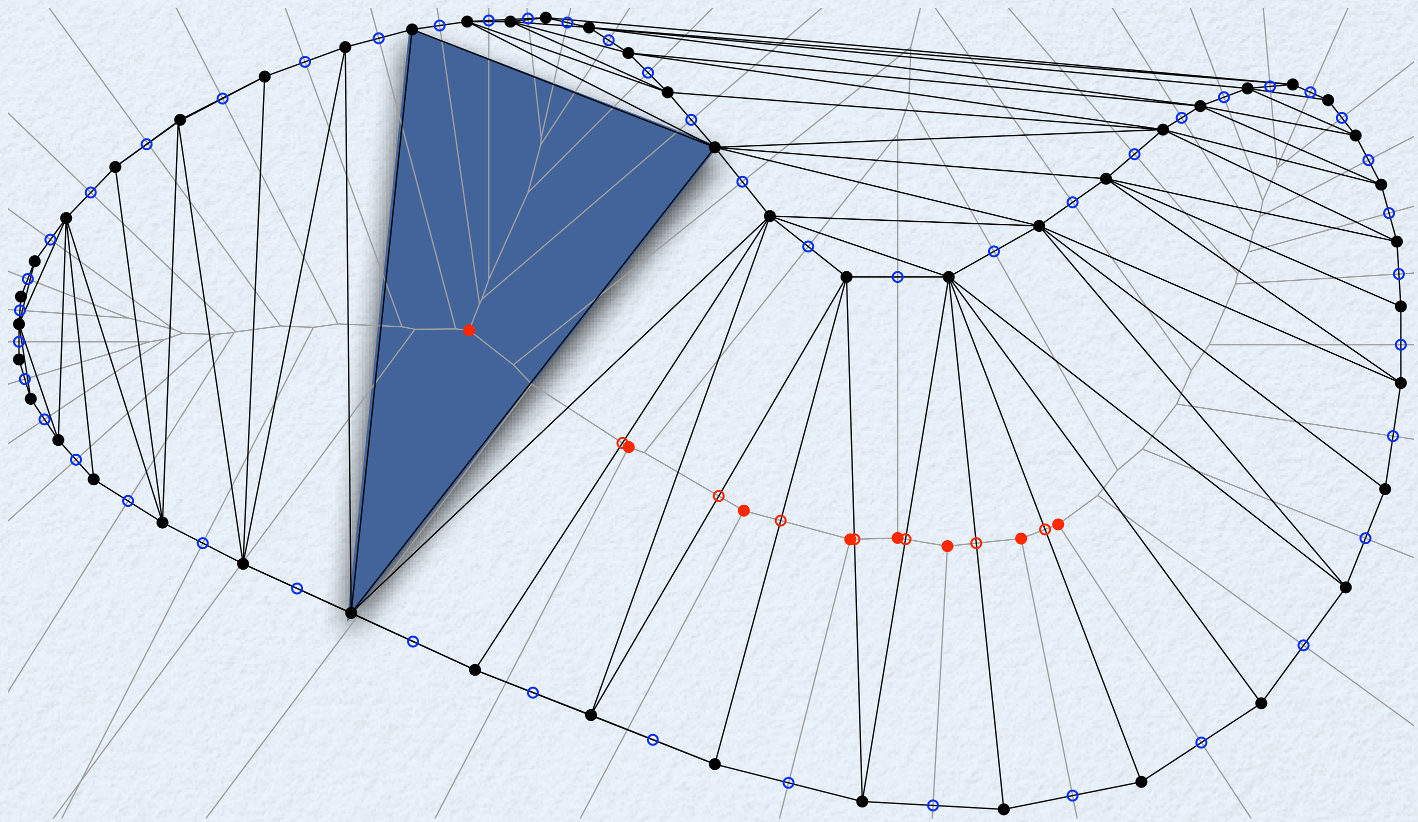
A point c is critical iff $\{c\} = V(c) \cap D(c)$.



Critical Points of Distance Function

A point c with $v(c) = 0$ is called critical.

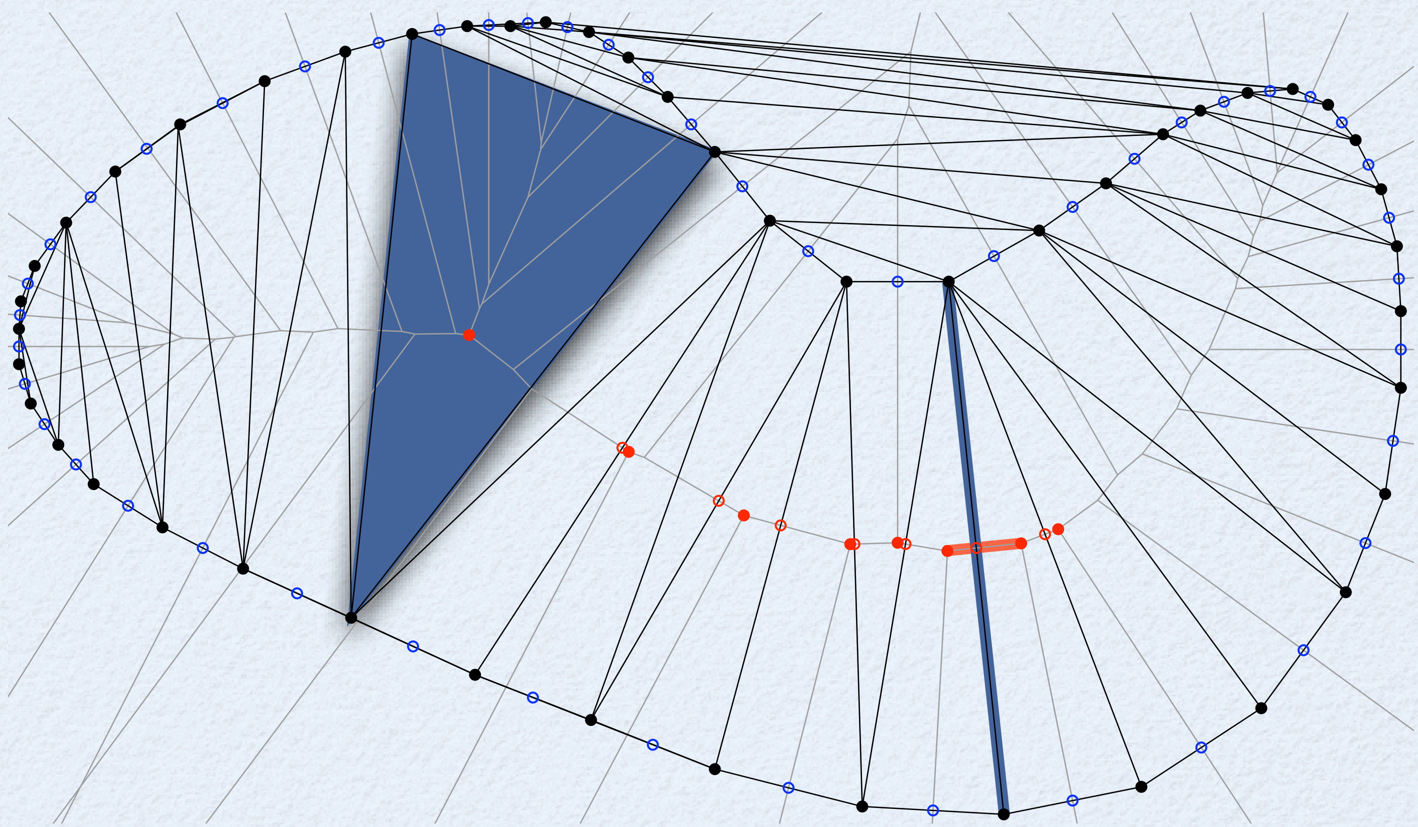
A point c is critical iff $\{c\} = V(c) \cap D(c)$.



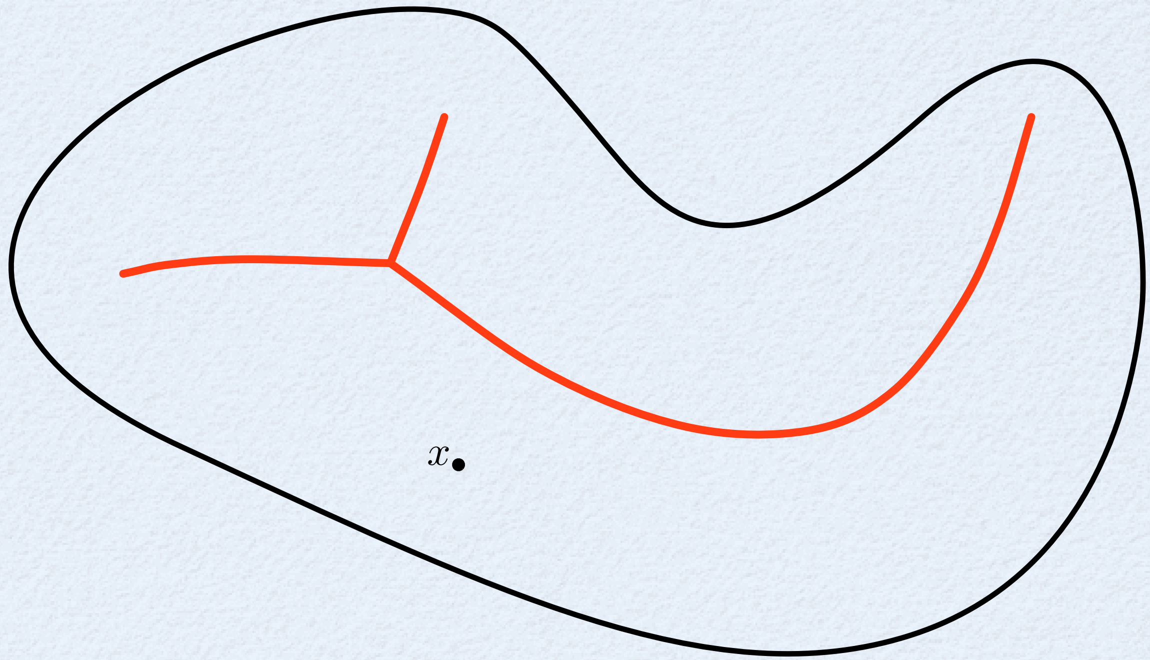
Critical Points of Distance Function

A point c with $v(c) = 0$ is called critical.

A point c is critical iff $\{c\} = V(c) \cap D(c)$.



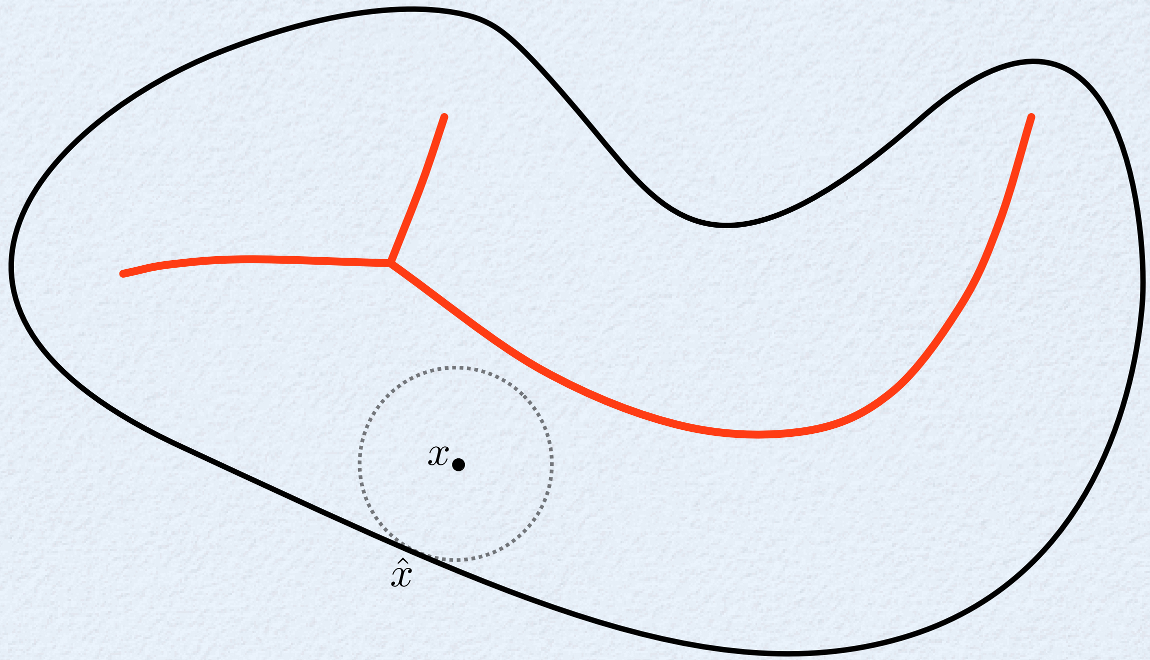
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$\begin{aligned} \Sigma_\delta &= \left\{ x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x}) \right\} \\ M_\delta &= \left\{ x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x}) \right\} \end{aligned}$$

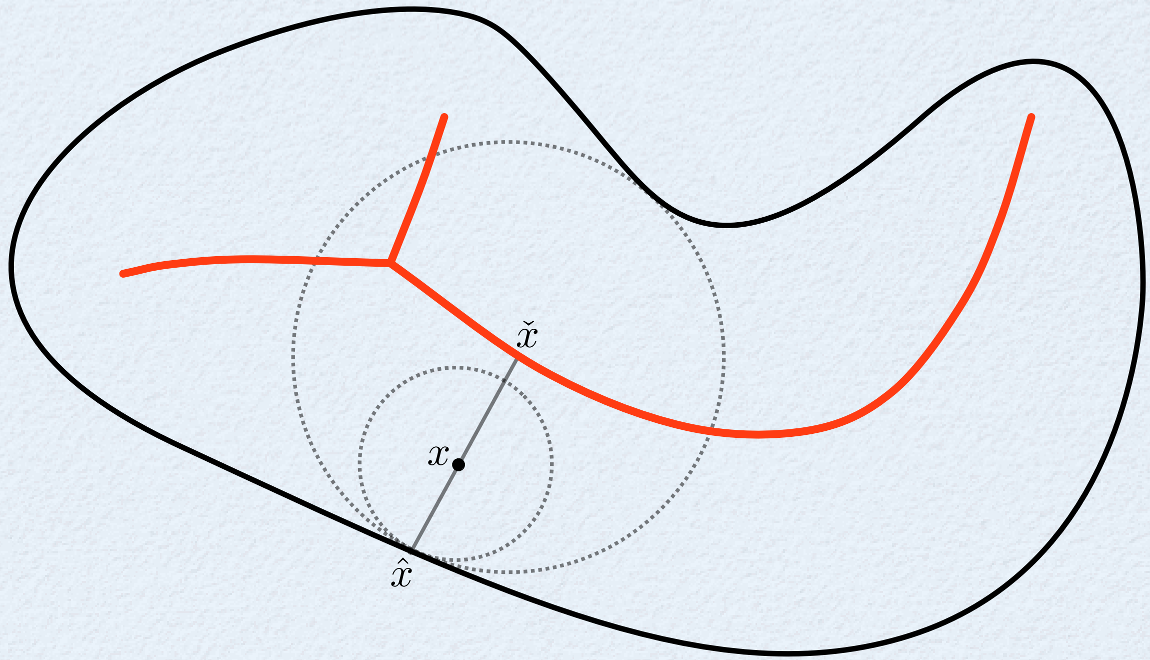
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$\begin{aligned} \Sigma_\delta &= \left\{ x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x}) \right\} \\ M_\delta &= \left\{ x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x}) \right\} \end{aligned}$$

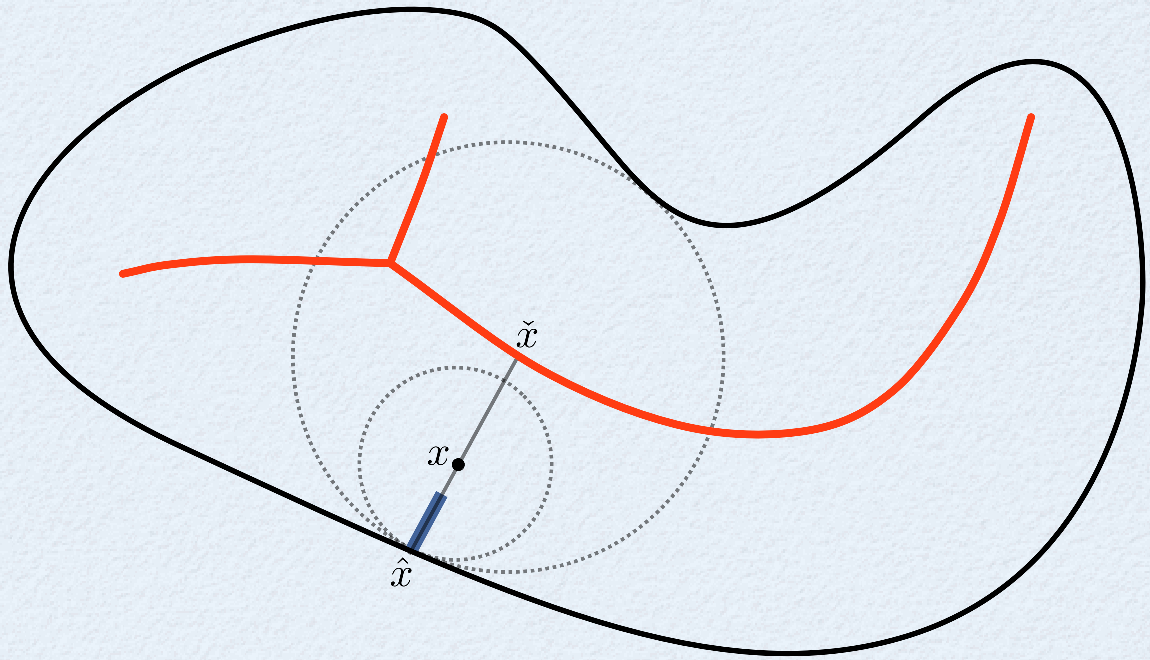
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$\begin{aligned} \Sigma_\delta &= \{ x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x}) \} \\ M_\delta &= \{ x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x}) \} \end{aligned}$$

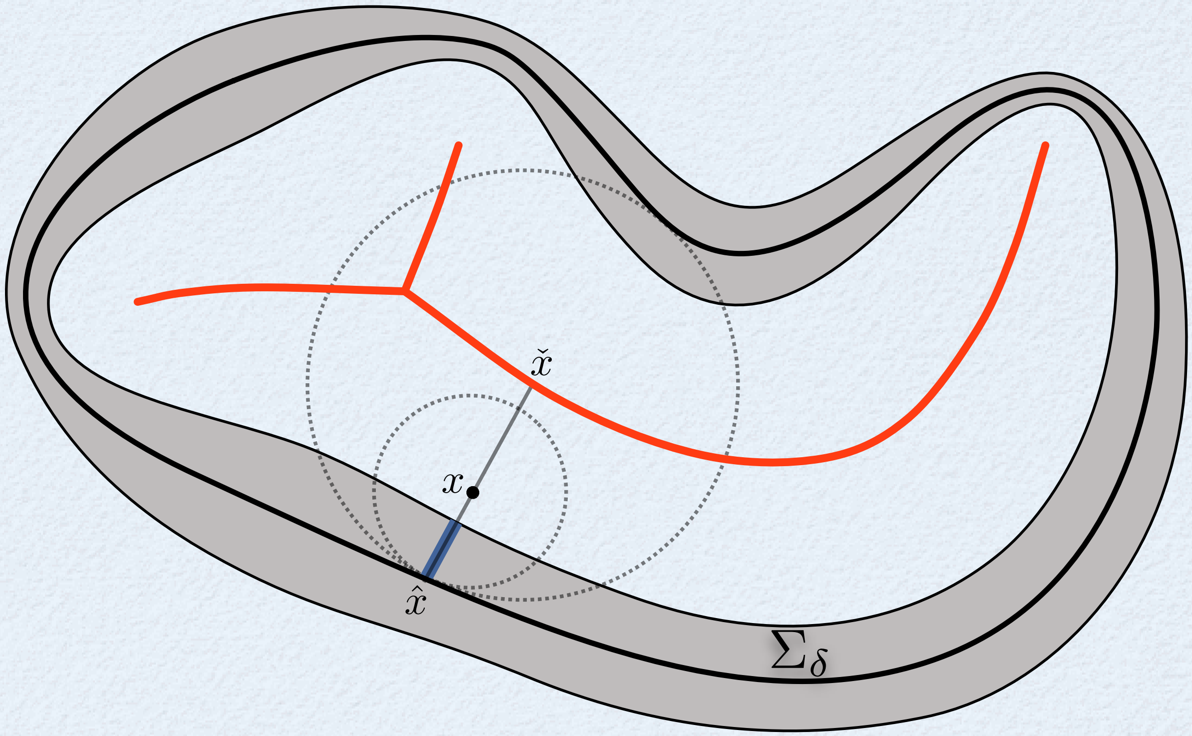
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$\begin{aligned} \Sigma_\delta &= \left\{x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x})\right\} \\ M_\delta &= \left\{x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x})\right\} \end{aligned}$$

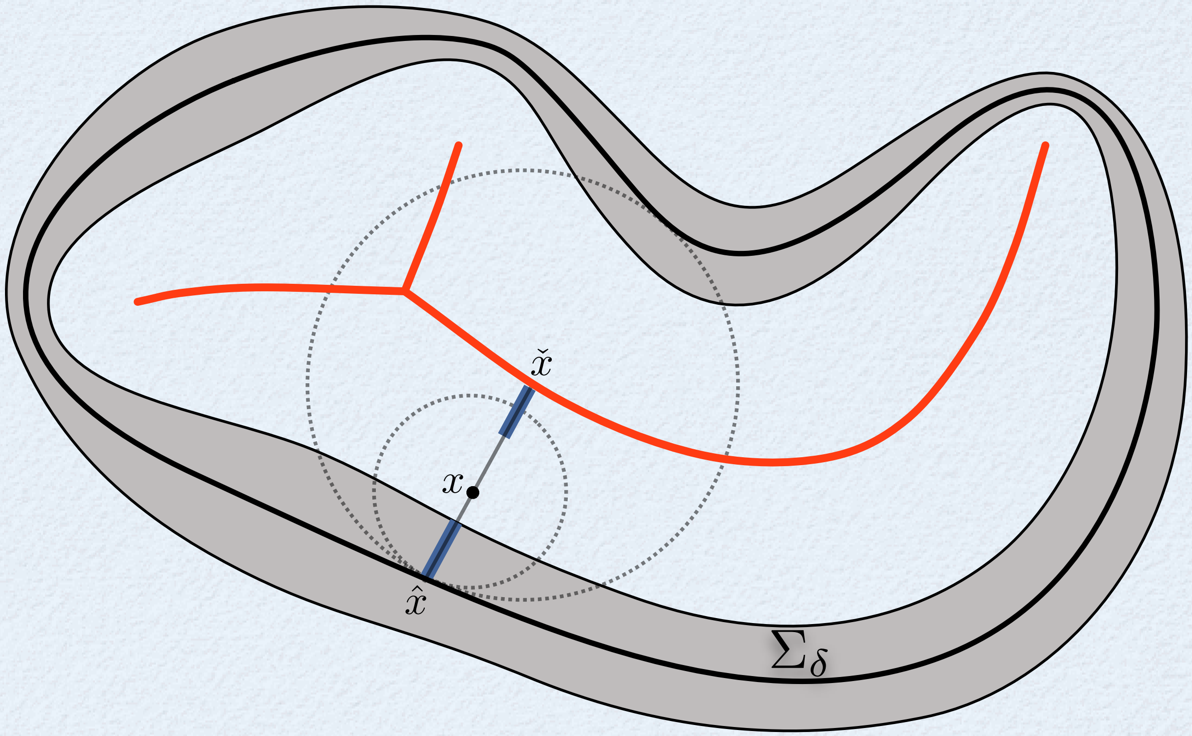
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$
\begin{aligned}
\Sigma_\delta &= \left\{ x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x}) \right\} \\
M_\delta &= \left\{ x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x}) \right\}
\end{aligned}
$$

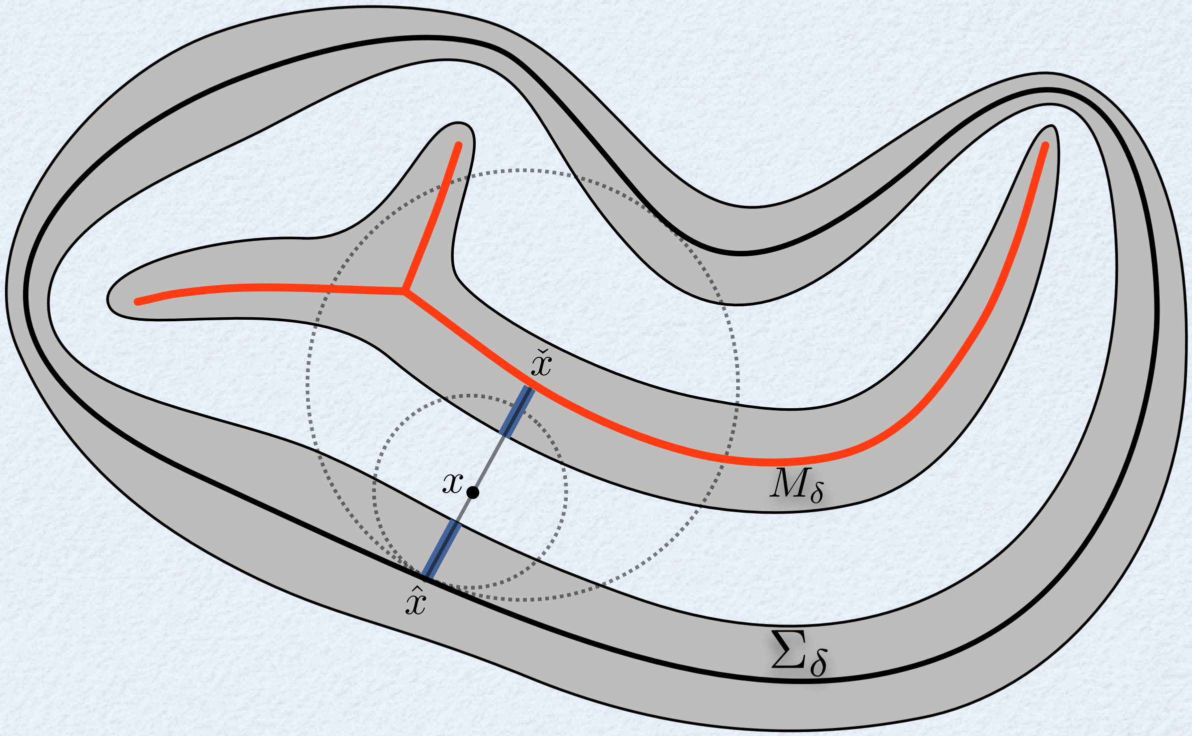
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$\begin{aligned} \Sigma_\delta &= \left\{ x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x}) \right\} \\ M_\delta &= \left\{ x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x}) \right\} \end{aligned}$$

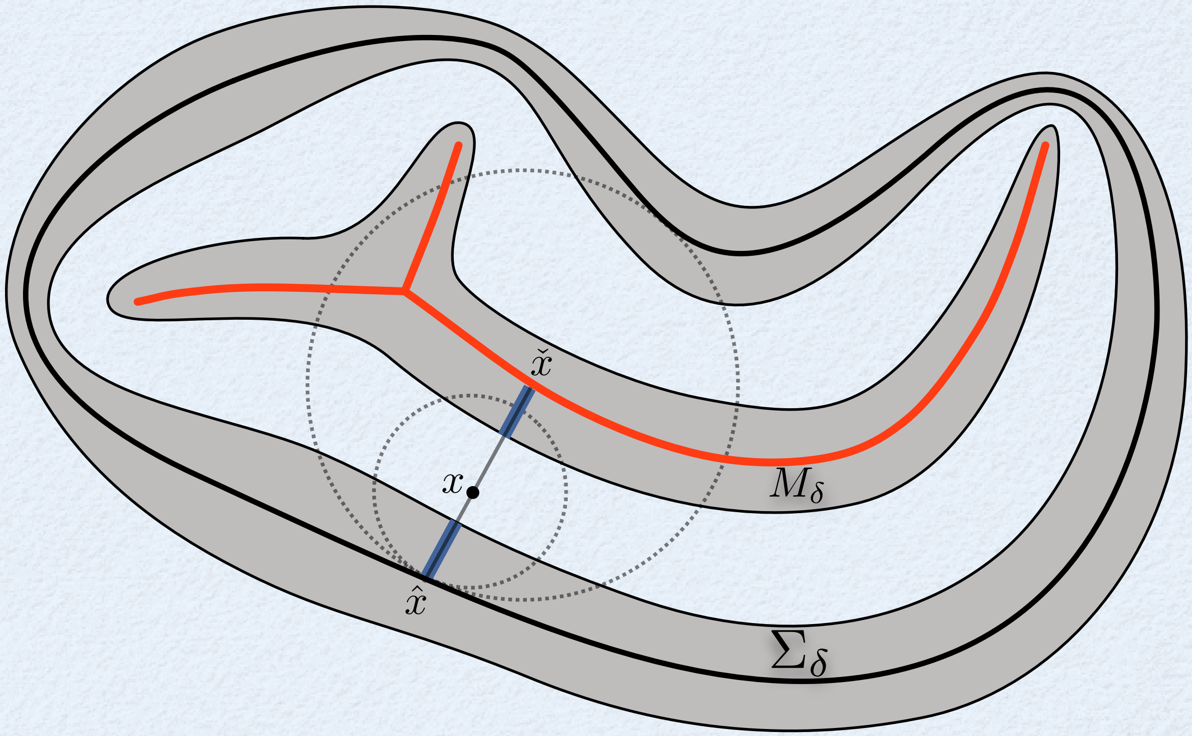
Separation of Critical Points



The δ-tubular neighborhoods of Σ and M:

$$\begin{aligned} \Sigma_\delta &= \{x \in \mathbb{R}^n \setminus M : \|x - \hat{x}\| < \delta f(\hat{x})\} \\ M_\delta &= \{x \in \mathbb{R}^n \setminus \Sigma : \|x - \check{x}\| < \delta f(\hat{x})\} \end{aligned}$$

Separation of Critical Points

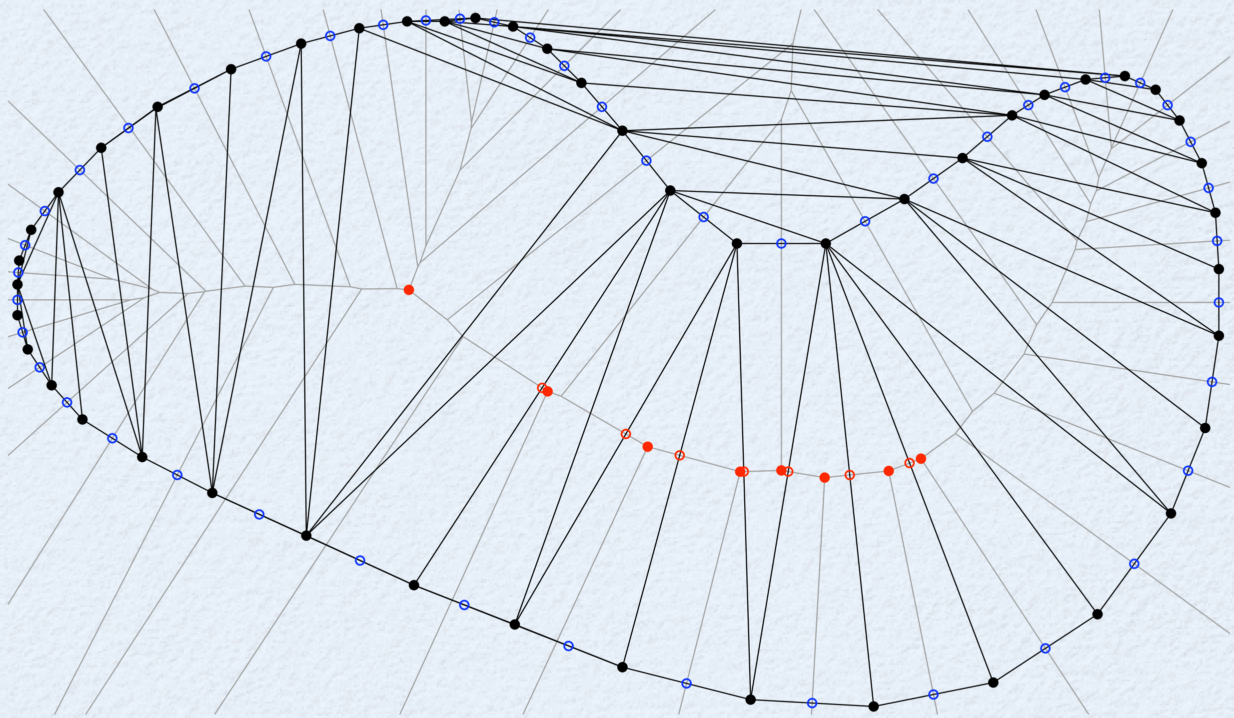


Theorem [DGRS'05]
If h is induced by an ε-sample of Σ with $\varepsilon < 1/\sqrt{3}$, the all critical points of h are contained in either Σ_{ε^2} or $M_{2\varepsilon^2}$.

Stable Manifold of a Critical Point

Stable manifold of a critical point c is everything that flows into c.

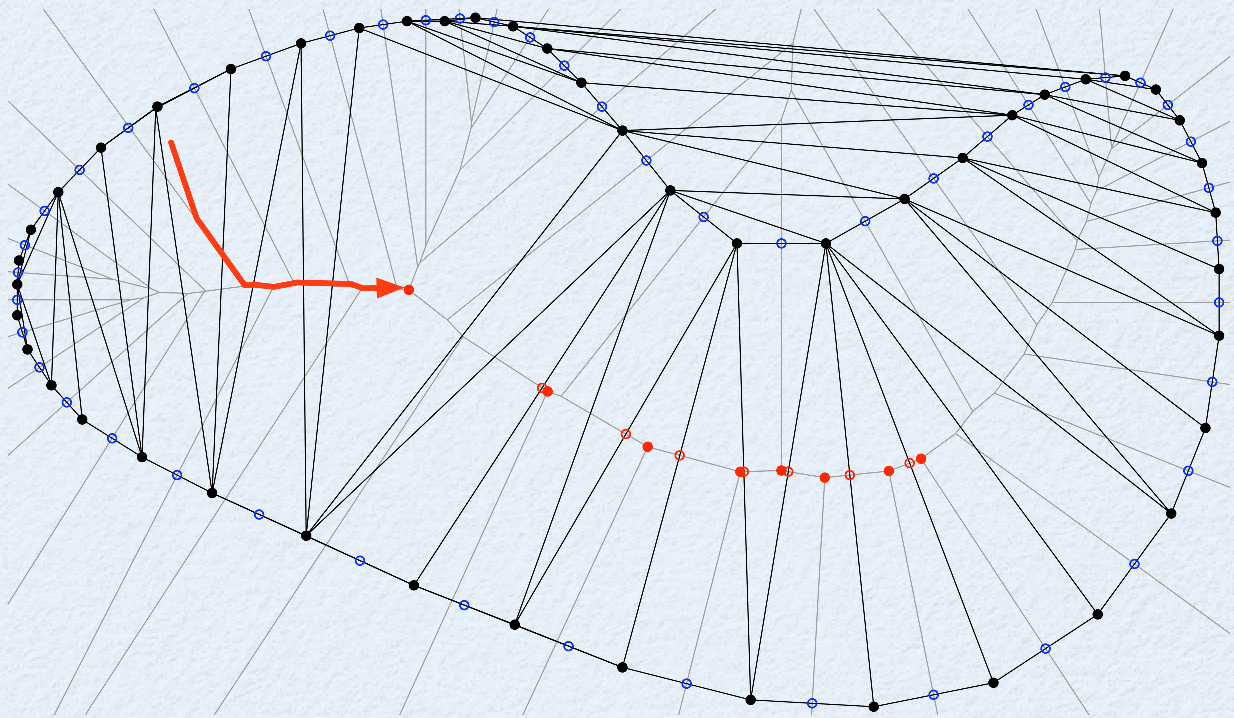
$$\mathsf{Sm}(c) = \{x : \phi(\infty, x) = c\}.$$



Stable Manifold of a Critical Point

Stable manifold of a critical point c is everything that flows into c.

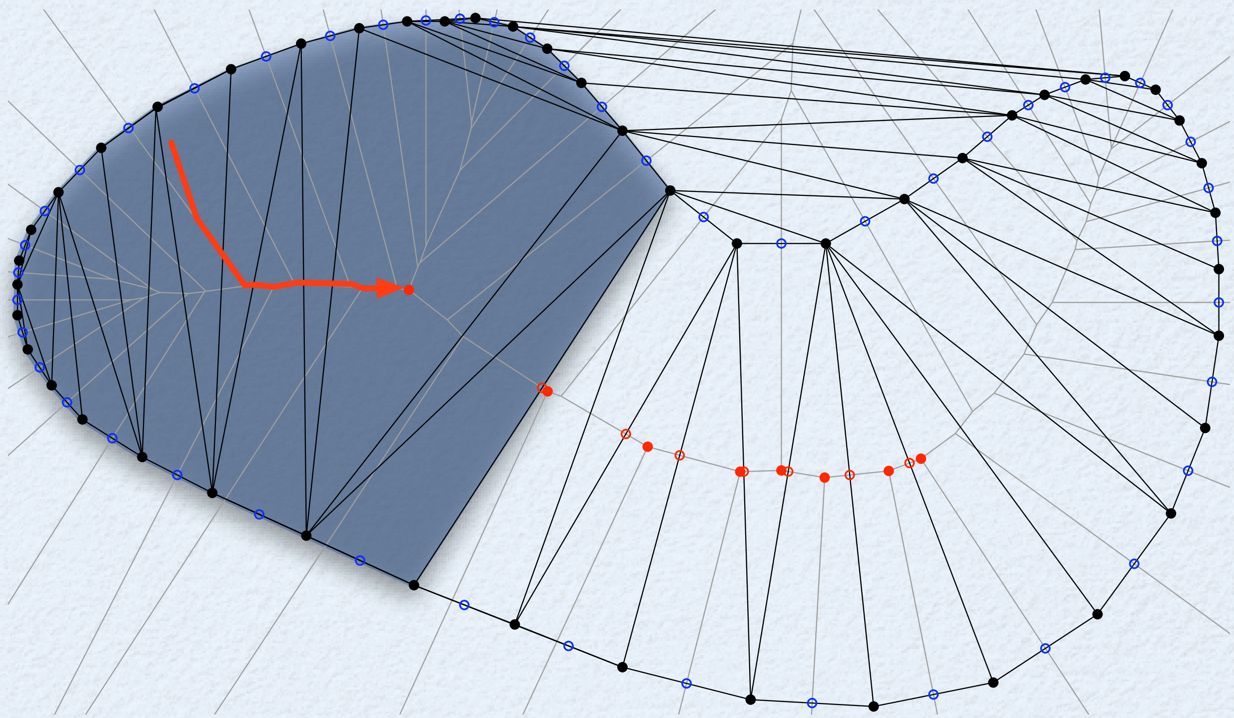
$$\mathsf{Sm}(c) = \{x : \phi(\infty, x) = c\}.$$



Stable Manifold of a Critical Point

Stable manifold of a critical point c is everything that flows into c.

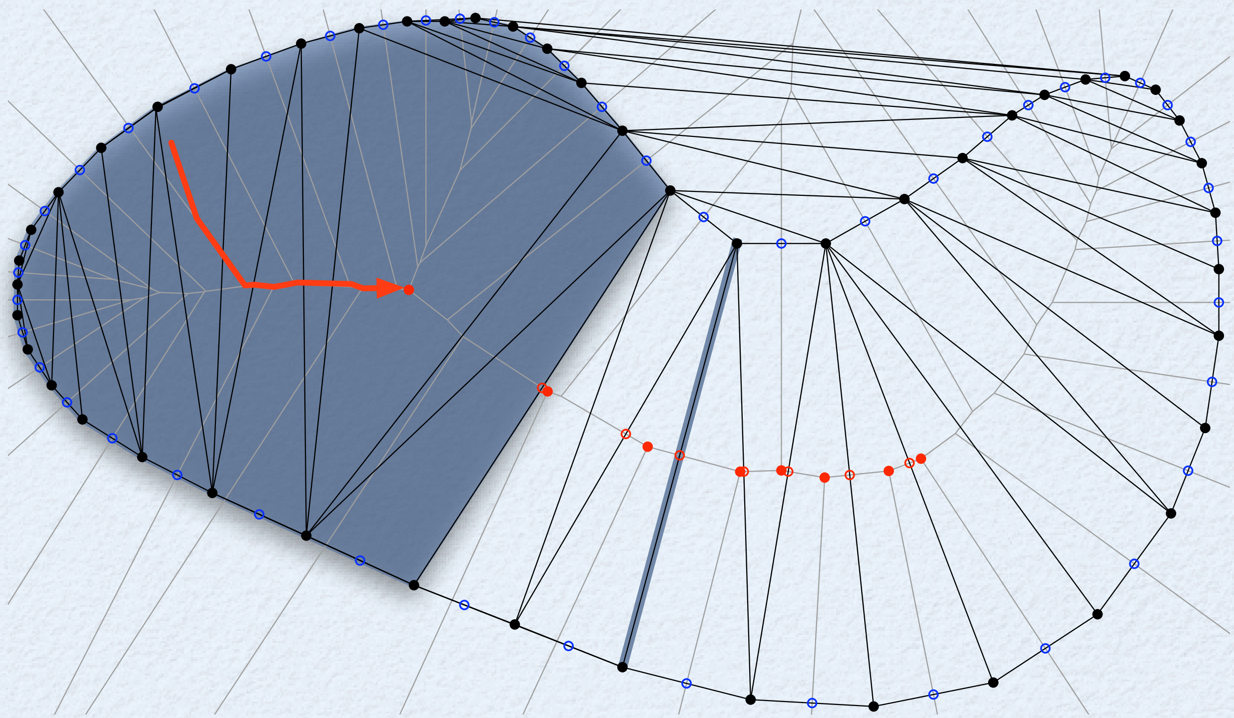
$$\mathsf{Sm}(c) = \{x : \phi(\infty, x) = c\}.$$



Stable Manifold of a Critical Point

Stable manifold of a critical point c is everything that flows into c.

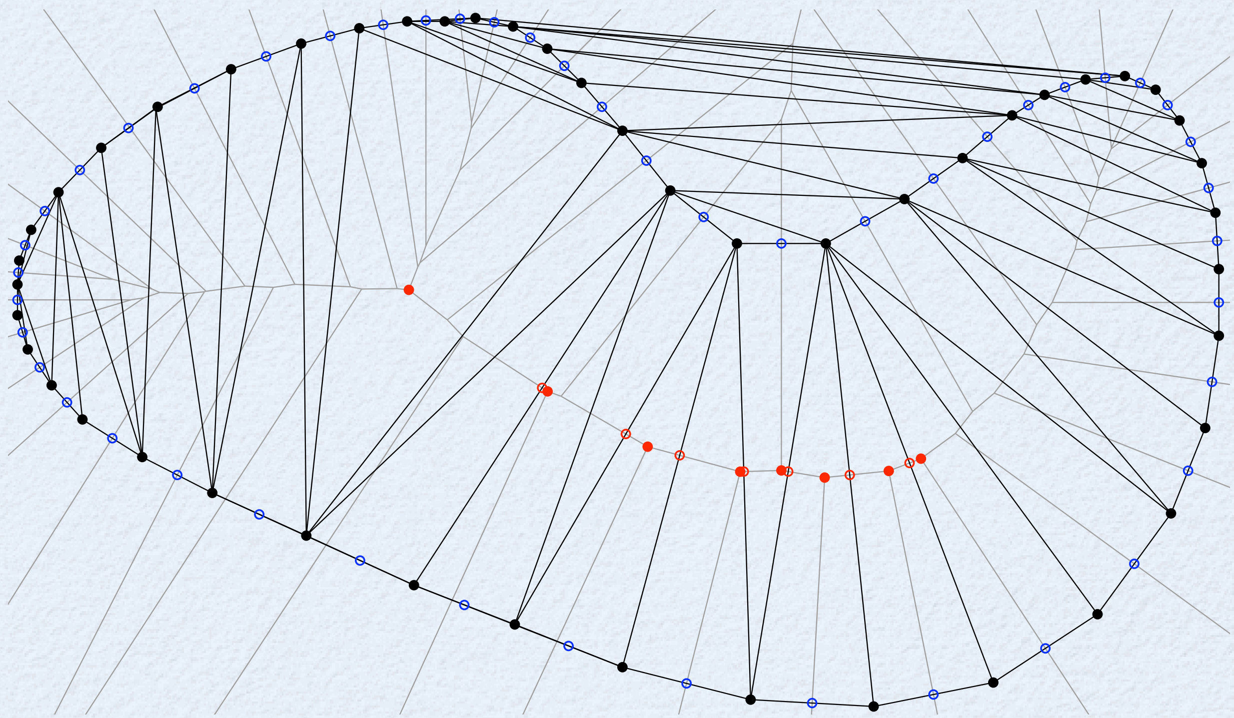
$$\mathsf{Sm}(c) = \{x : \phi(\infty, x) = c\}.$$



Unstable Manifold of a Critical Point

Unstable manifold of a critical point c is everything that flows "out of" c.

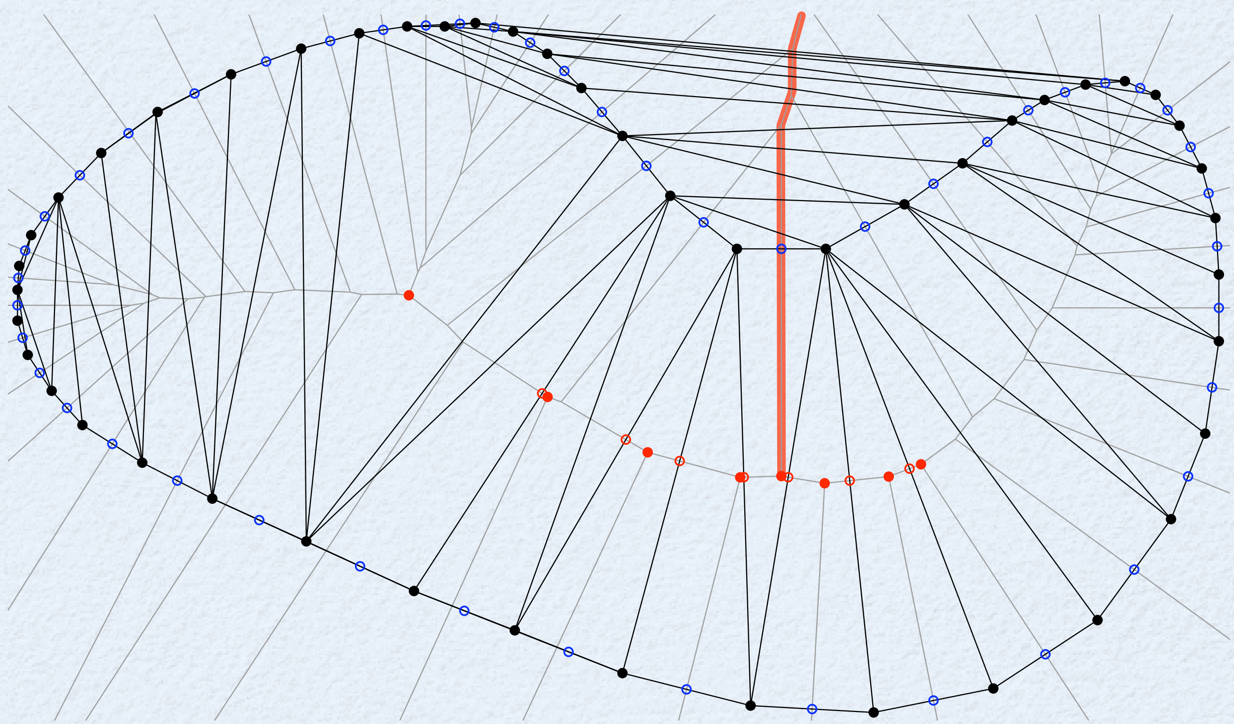
$$\mathsf{Um}(c) = \bigcap_{\varepsilon > 0} \phi(B(x, \varepsilon)) = \phi(V(c)).$$



Unstable Manifold of a Critical Point

Unstable manifold of a critical point c is everything that flows "out of" c.

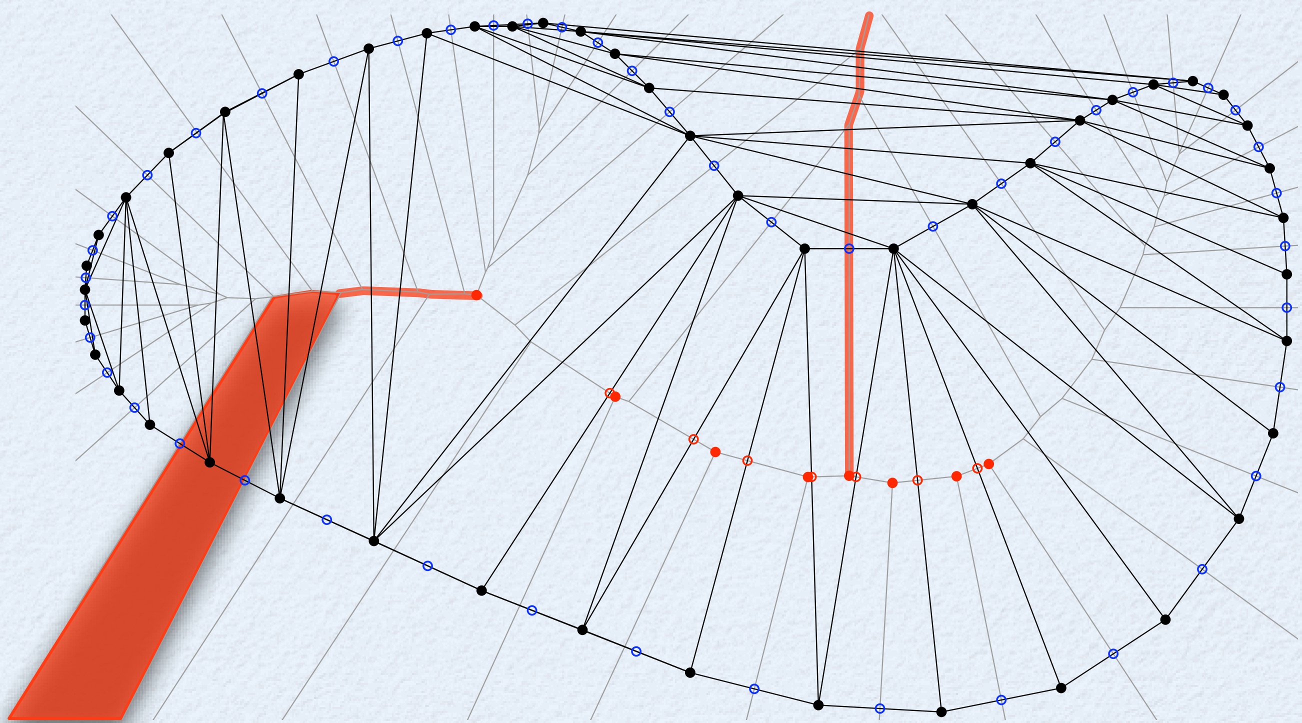
$$\mathsf{Um}(c) = \bigcap_{\varepsilon > 0} \phi(B(x, \varepsilon)) = \phi(V(c)).$$



Unstable Manifold of a Critical Point

Unstable manifold of a critical point c is everything that flows "out of" c.

$$\mathrm{Um}(c) = \bigcap_{\varepsilon > 0} \phi(B(x, \varepsilon)) = \phi(V(c)).$$



A Criterion for Homotopy Equivalence

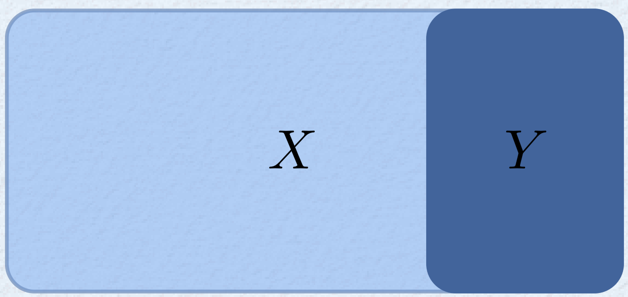
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : [0,1] \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0,x) = x$
2. $\forall y \in Y, \forall t \in [0,1] : H(t,y) \in Y$
3. $\forall x \in X : H(1,x) \in Y$

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

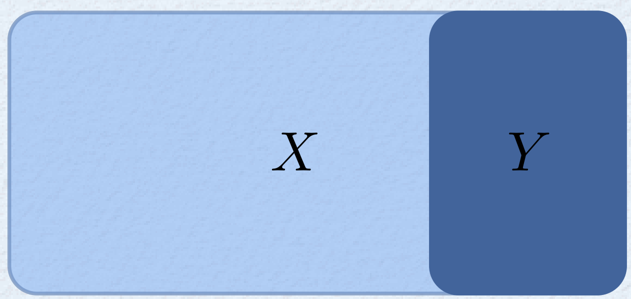
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : \underbrace{[0,1]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0,x) = x$
2. $\forall y \in Y, \forall t \in [0,1] : H(t,y) \in Y$
3. $\forall x \in X : H(1,x) \in Y$

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

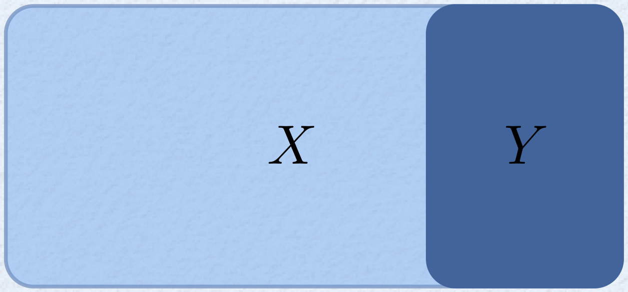
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : \underbrace{[0,1]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0,x) = x$ Identity at time 0
2. $\forall y \in Y, \forall t \in [0,1] : H(t,y) \in Y$
3. $\forall x \in X : H(1,x) \in Y$

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

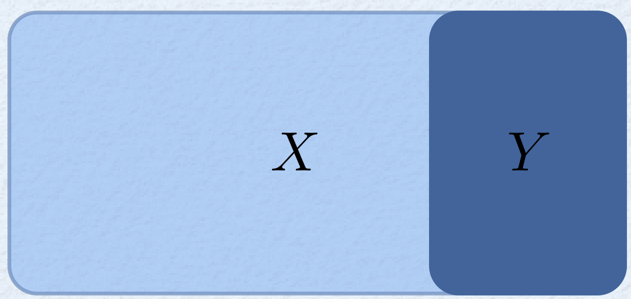
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : \underbrace{[0,1]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0,x) = x$ Identity at time 0
2. $\forall y \in Y, \forall t \in [0,1] : H(t,y) \in Y$ Nothing leaves Y
3. $\forall x \in X : H(1,x) \in Y$

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

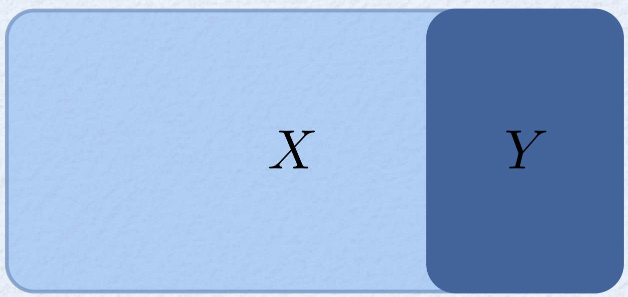
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : \underbrace{[0,1]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0,x) = x$ — Identity at time 0
2. $\forall y \in Y, \forall t \in [0,1] : H(t,y) \in Y$ — Nothing leaves Y
3. $\forall x \in X : H(1,x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

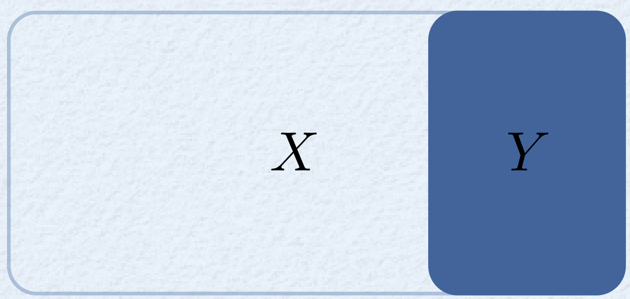
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : \underbrace{[0,1]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0,x) = x$ — Identity at time 0
2. $\forall y \in Y, \forall t \in [0,1] : H(t,y) \in Y$ — Nothing leaves Y
3. $\forall x \in X : H(1,x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

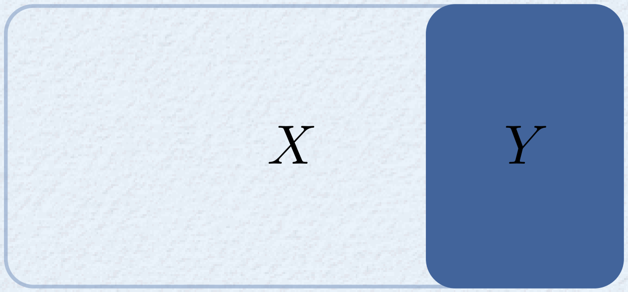
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$H : \underbrace{[0, T]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : H(0, x) = x$ — Identity at time 0
2. $\forall y \in Y, \forall t \in [0, T] : H(t, y) \in Y$ — Nothing leaves Y
3. $\forall x \in X : H(T, x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

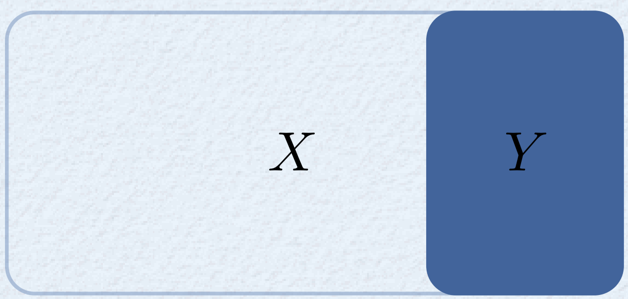
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$\phi : \underbrace{[0, T]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : \phi(0, x) = x$ — Identity at time 0
2. $\forall y \in Y, \forall t \in [0, T] : \phi(t, y) \in Y$ — Nothing leaves Y
3. $\forall x \in X : \phi(T, x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

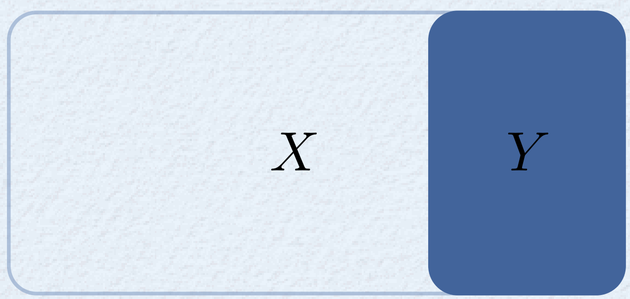
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$\phi : \underbrace{[0, T]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : \phi(0, x) = x$ — Always true
2. $\forall y \in Y, \forall t \in [0, T] : \phi(t, y) \in Y$ — Nothing leaves Y
3. $\forall x \in X : \phi(T, x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

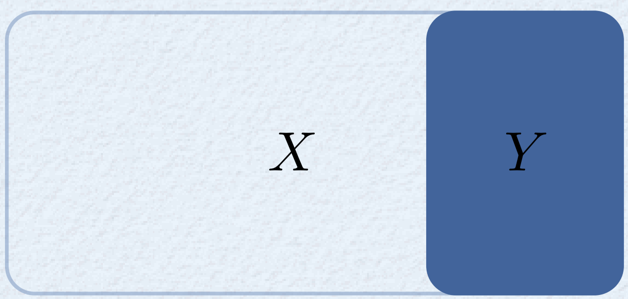
Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

$$\phi : \underbrace{[0, T]}_{\text{time}} \times X \to X$$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : \phi(0, x) = x$ — Always true
2. $\forall y \in Y, \forall t \in [0, T] : \phi(t, y) \in Y$ — $\phi(Y) = Y$
3. $\forall x \in X : \phi(T, x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

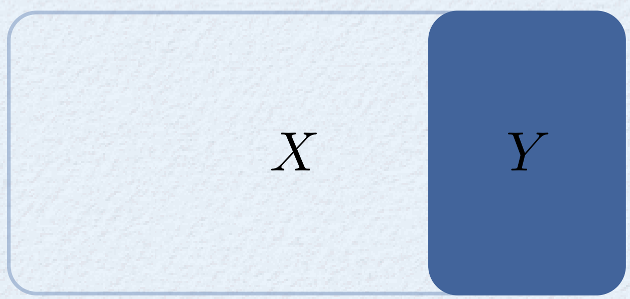
$$\phi : \underbrace{[0, T]}_{\text{time}} \times X \to X$$

$\phi(X) = X$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : \phi(0, x) = x$ — Always true
2. $\forall y \in Y, \forall t \in [0, T] : \phi(t, y) \in Y$ — $\phi(Y) = Y$
3. $\forall x \in X : \phi(T, x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



A Criterion for Homotopy Equivalence

Proposition. Let X and $Y \subseteq X$ be arbitrary sets and

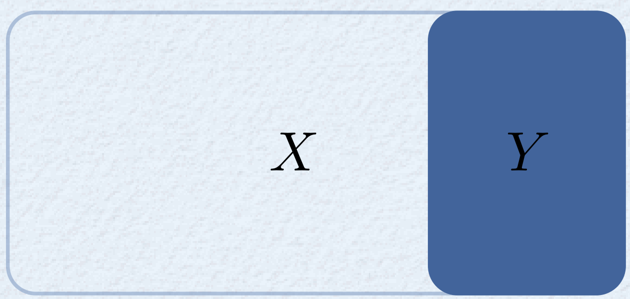
$$\phi : \underbrace{[0,T]}_{\text{time}} \times X \to X$$

$\phi(X) = X$

be a continuous function (on both variables) satisfying

1. $\forall x \in X : \phi(0, x) = x$ — Always true
2. $\forall y \in Y, \forall t \in [0,T] : \phi(t, y) \in Y$ — $\phi(Y) = Y$
3. $\forall x \in X : \phi(T, x) \in Y$ — Everything in Y by time 1

Then X and Y have the same homotopy type.



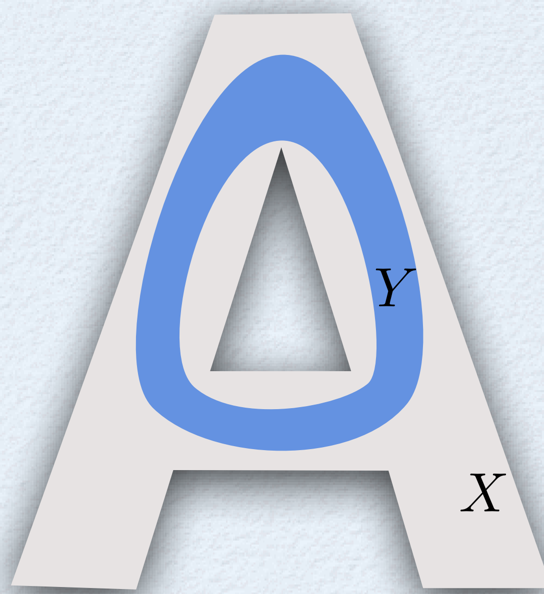
This is the idea Lieutier used in [Lieuteir'04] to show $M(S) \simeq S$.

In Other Words ...

Key Theorem. If $Y \subset X$ are bounded and

1. $\phi(X) = X$ and $\phi(Y) = Y$, and
2. $\|v(x)\| \geq c > 0$ for $x \in X \setminus Y$,

then X and Y are homotopy equivalent.



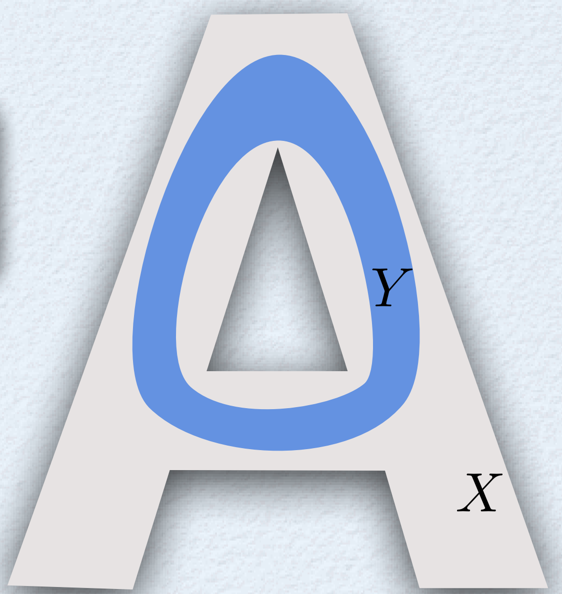
In Other Words ...

Key Theorem. If $Y \subset X$ are bounded and

1. $\phi(X) = X$ and $\phi(Y) = Y$, and
2. $\|v(x)\| \geq c > 0$ for $x \in X \setminus Y$,

then X and Y are homotopy equivalent.

X and Y are flow-tight



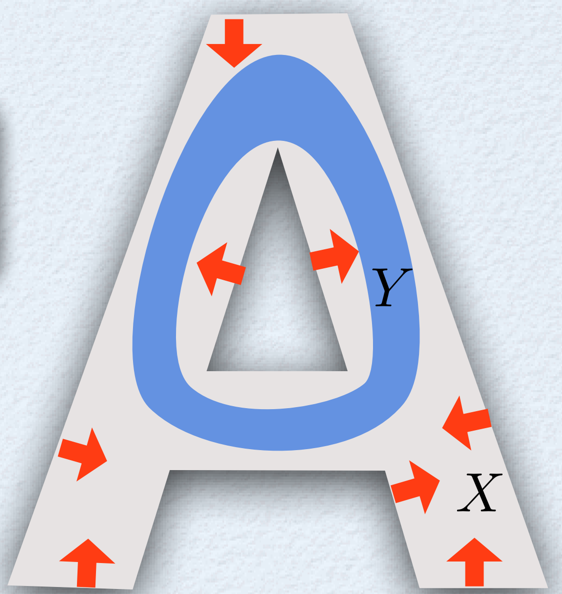
In Other Words ...

Key Theorem. If $Y \subset X$ are bounded and

1. $\phi(X) = X$ and $\phi(Y) = Y$, and
2. $\|v(x)\| \geq c > 0$ for $x \in X \setminus Y$,

then X and Y are homotopy equivalent.

X and Y are flow-tight



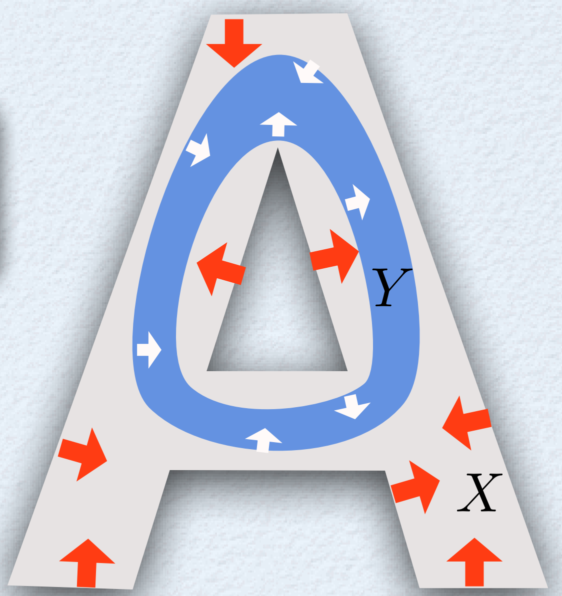
In Other Words ...

Key Theorem. If $Y \subset X$ are bounded and

1. $\phi(X) = X$ and $\phi(Y) = Y$, and
2. $\|v(x)\| \geq c > 0$ for $x \in X \setminus Y$,

then X and Y are homotopy equivalent.

X and Y are flow-tight



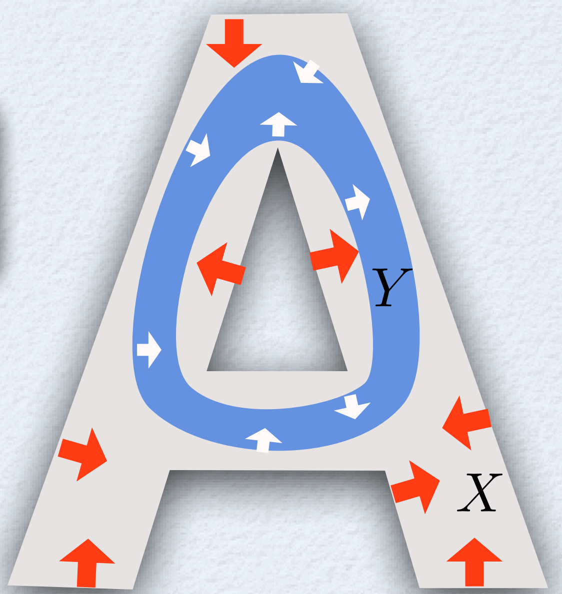
In Other Words ...

Key Theorem. If $Y \subset X$ are bounded and

1. $\phi(X) = X$ and $\phi(Y) = Y$, and
2. $\|v(x)\| \geq c > 0$ for $x \in X \setminus Y$,

then X and Y are homotopy equivalent.

X and Y are flow-tight



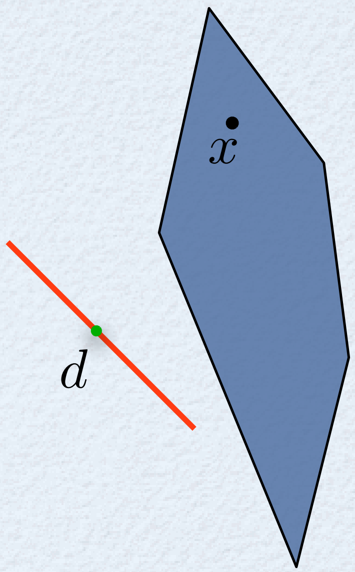
Proof. If $\phi(t,x) \notin Y$, then

$$\begin{aligned} h(\phi(t,x)) &= h(x) + \int_0^t \|v(\phi(\tau,x))\|^2 d\tau \\ &\geq h(x) + \int_0^t c^2 d\tau \\ &= h(x) + tc^2 \\ &< d_H(X,P)^2. \end{aligned}$$

A Handy Lower Bound for Speed

If $V(x) \cap D(x) = \emptyset$ then

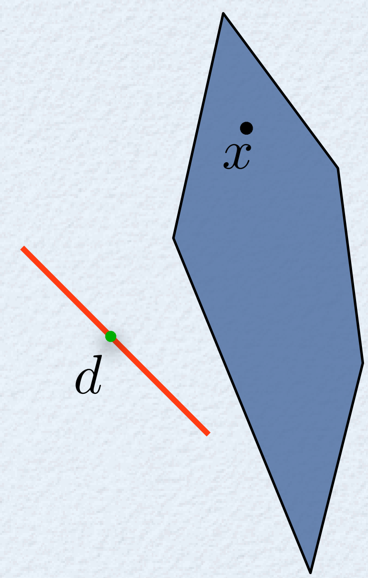
$$\begin{aligned} \|v(x)\| &= 2 \cdot \|x - d(x)\| \\ &\geq 2 \cdot \text{dist}(V(x), D(x)). \end{aligned}$$



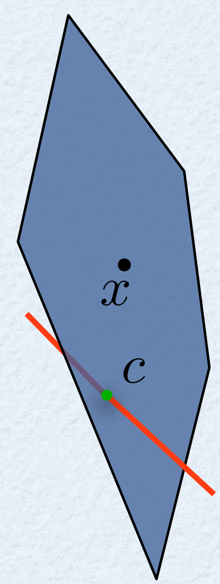
A Handy Lower Bound for Speed

If $V(x) \cap D(x) = \emptyset$ then

$$\begin{aligned} \|v(x)\| &= 2 \cdot \|x - d(x)\| \\ &\geq 2 \cdot \mathsf{dist}(V(x), D(x)). \end{aligned}$$



If $V(x) \cap D(x) = \{c\}$ then $x \in \mathsf{Um}(c)$.

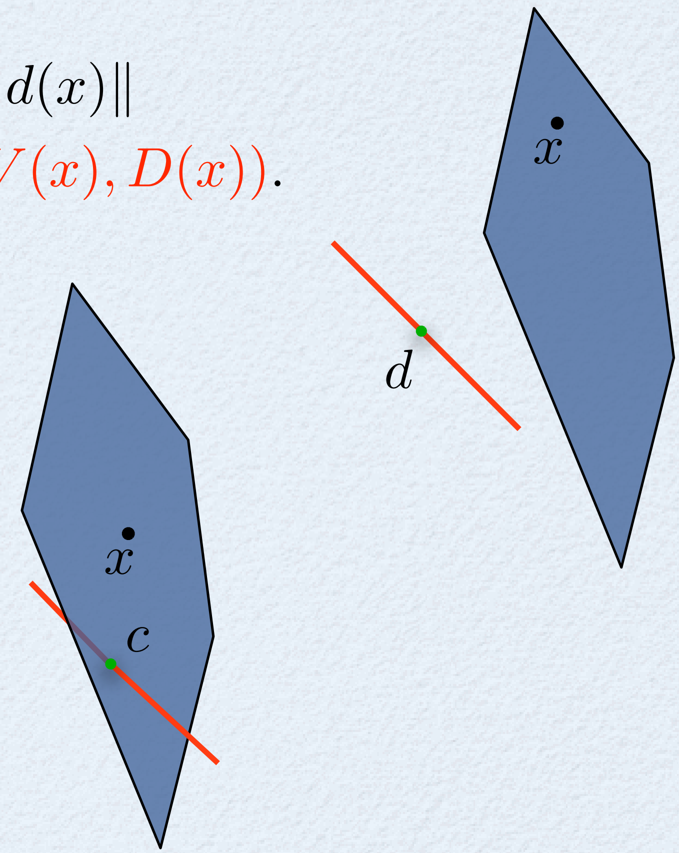


A Handy Lower Bound for Speed

If $V(x) \cap D(x) = \emptyset$ then

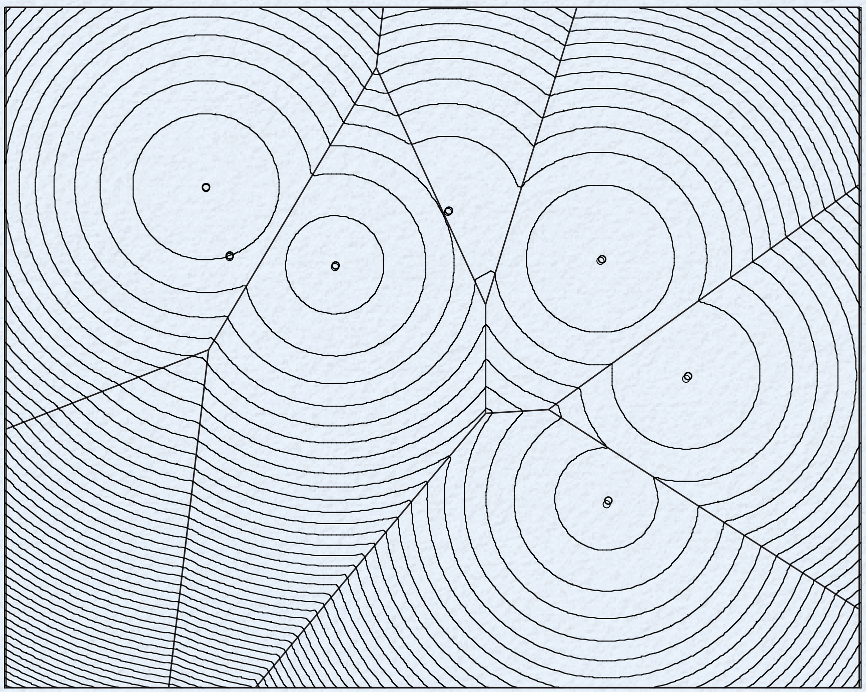
$$\begin{aligned} \|v(x)\| &= 2 \cdot \|x - d(x)\| \\ &\geq 2 \cdot \mathsf{dist}(V(x), D(x)). \end{aligned}$$

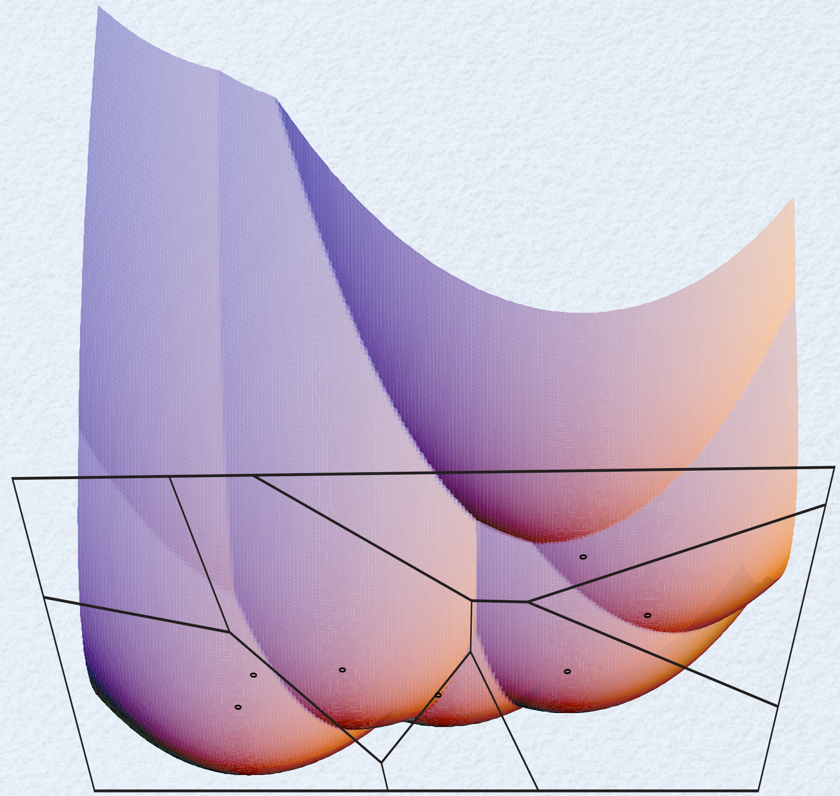
If $V(x) \cap D(x) = \{c\}$ then $x \in \mathsf{Um}(c)$.



So, if $\mathsf{Um}(c) \subset Y$ we are fine!

Flow Induced by Weighted Points





Squared distance to p with weight w_p is $\|x - p\|^2 - w_p$.

The squared distance to a set P of weighted points is

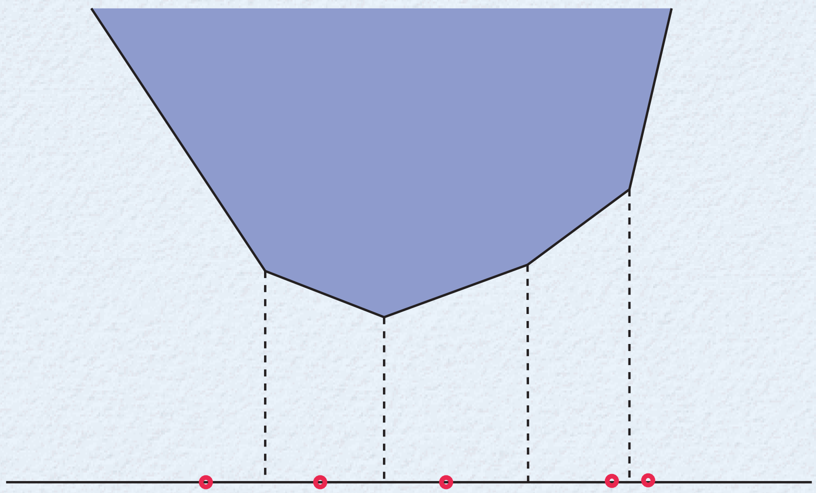
$$h(x) = \min_{p \in P} \|x - p\|^2 - w_p.$$

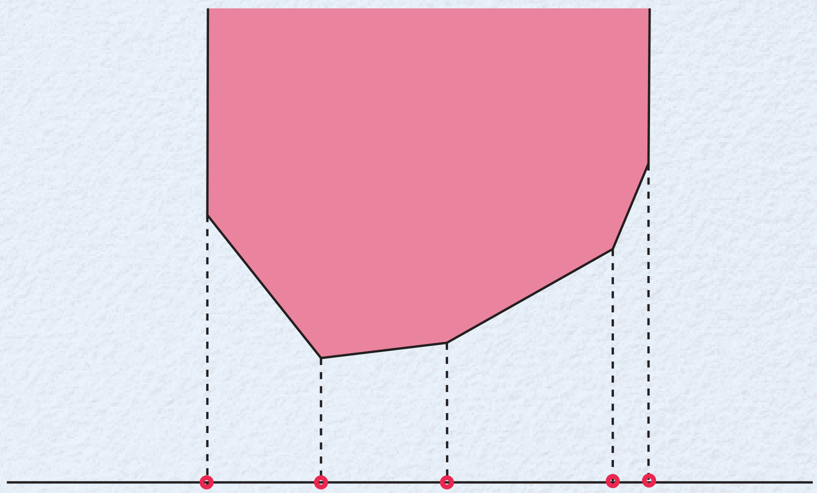
The
WRAP Algorithm

Polarity

For every set P of weighted points there is a set Q of weighted points such that

$$\text{Vor } P = \text{Del } Q \qquad \text{and} \qquad \text{Del } P = \text{Vor } Q$$

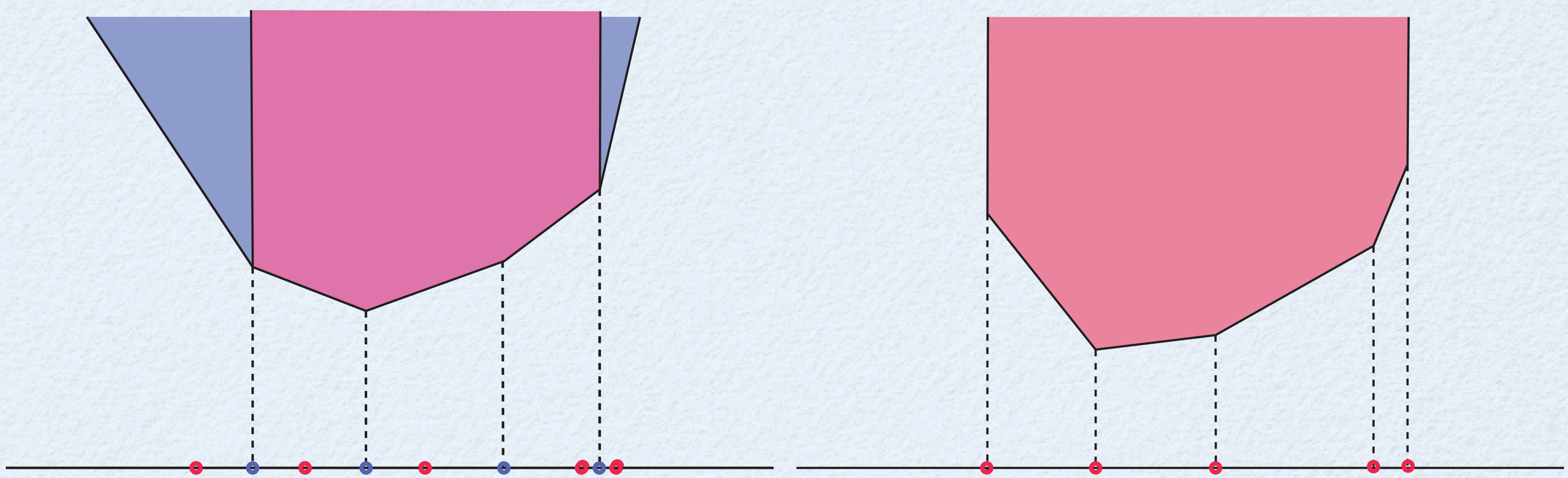




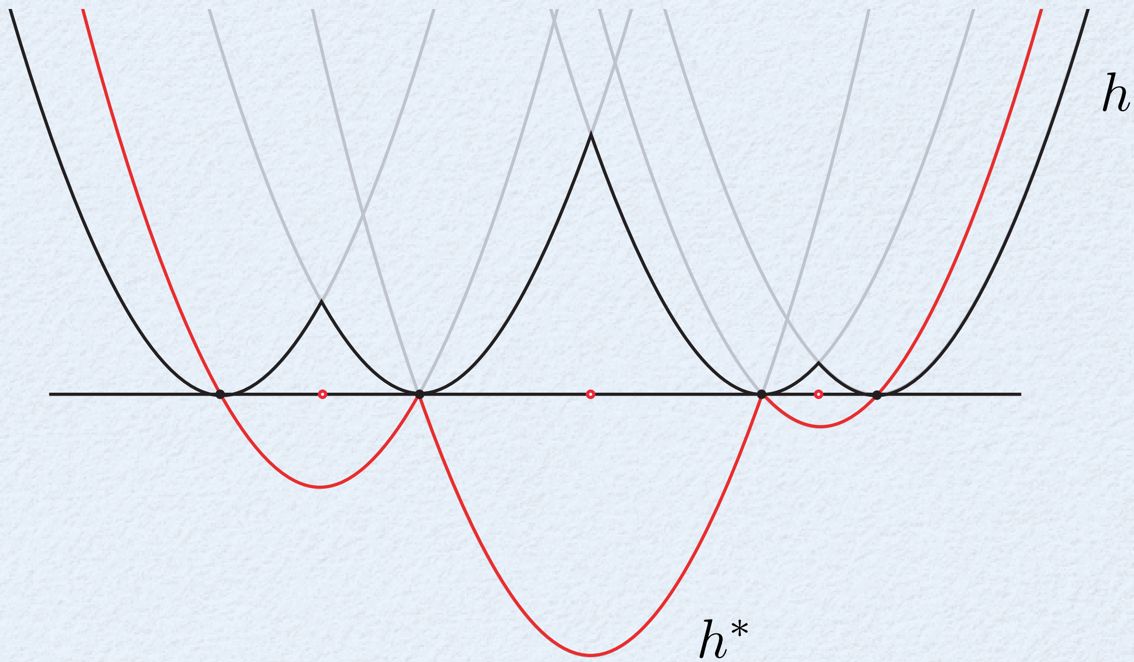
Polarity

For every set P of weighted points there is a set Q of weighted points such that

$$\text{Vor } P = \text{Del } Q \qquad \text{and} \qquad \text{Del } P = \text{Vor } Q$$



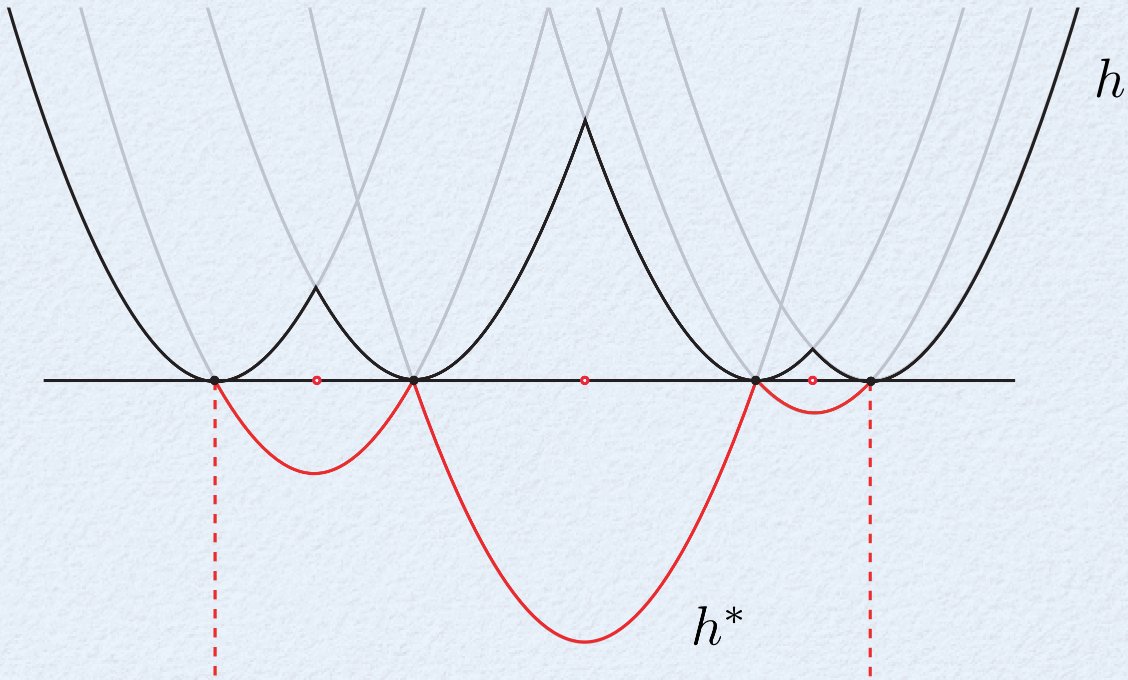
Voronoi Vertices as Weighted Points



For unweighted P, Q is the Voronoi vertices of P and for $q \in Q$:

$$w_q = \mathsf{dist}(q, P)^2.$$

Voronoi Vertices as Weighted Points

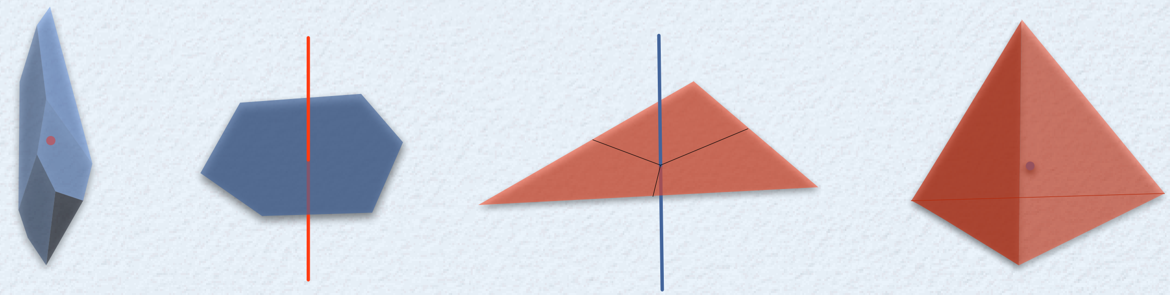


For unweighted P, Q is the Voronoi vertices of P and for $q \in Q$:

$$w_q = \mathrm{dist}(q, P)^2.$$

Critical Points of h^*

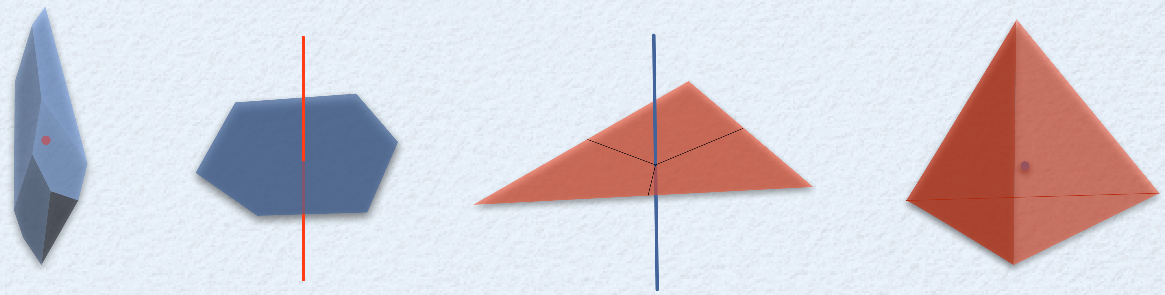
Observation. critical points of h^* and h are the same.



A simplex $\tau \in \mathrm{Del}\ P$ that contains a critical point is called a centered simplex.

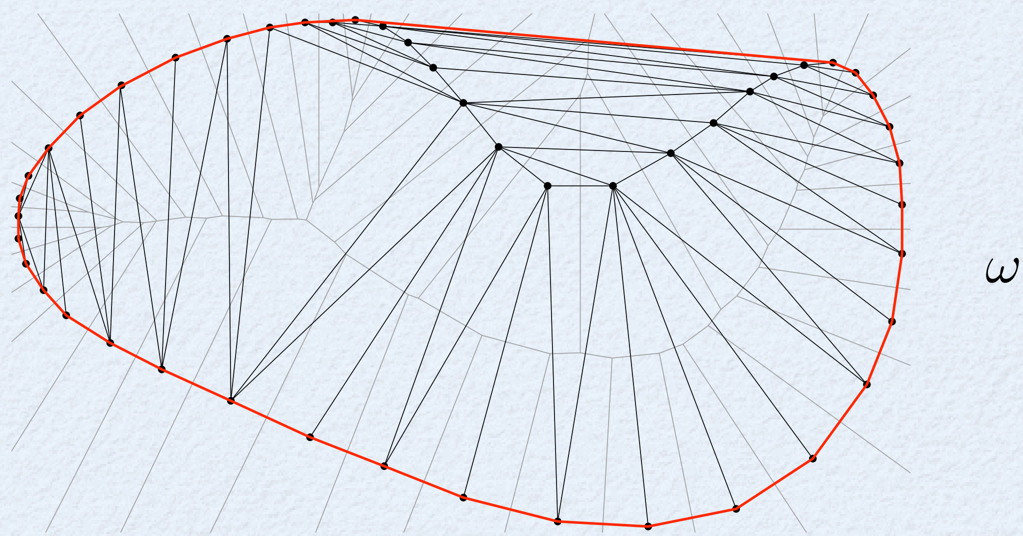
Critical Points of h^*

Observation. critical points of h^* and h are the same.



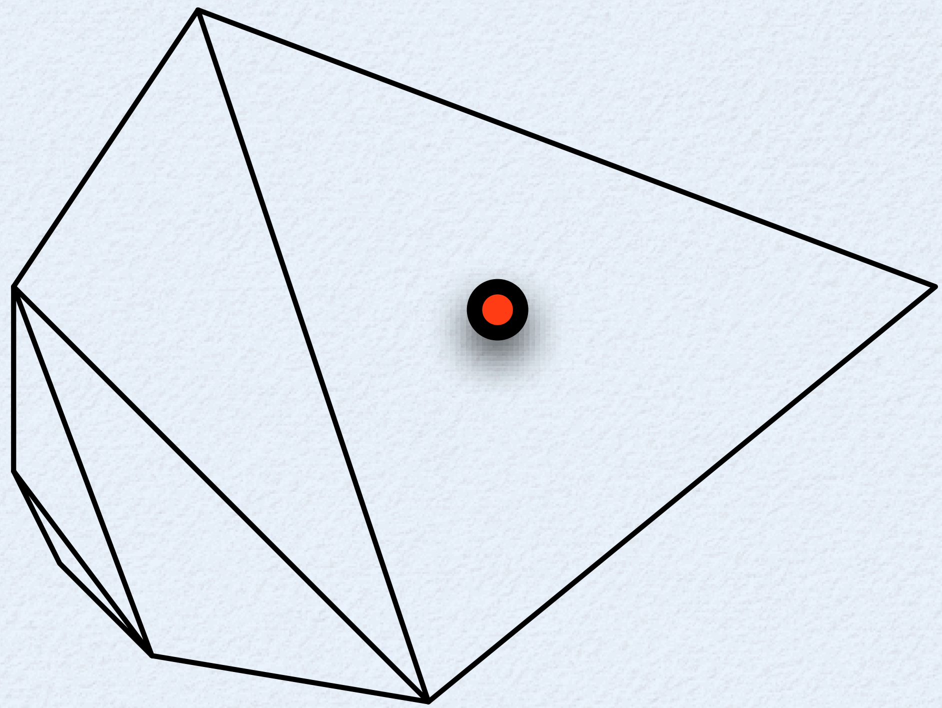
A simplex $\tau \in \mathrm{Del}\ P$ that contains a critical point is called a centered simplex.

We treat $\mathbb{R}^n \setminus \mathrm{conv}\ P$ as an abstract critical simplex ω.



A Partial Order on Delaunay Simplices

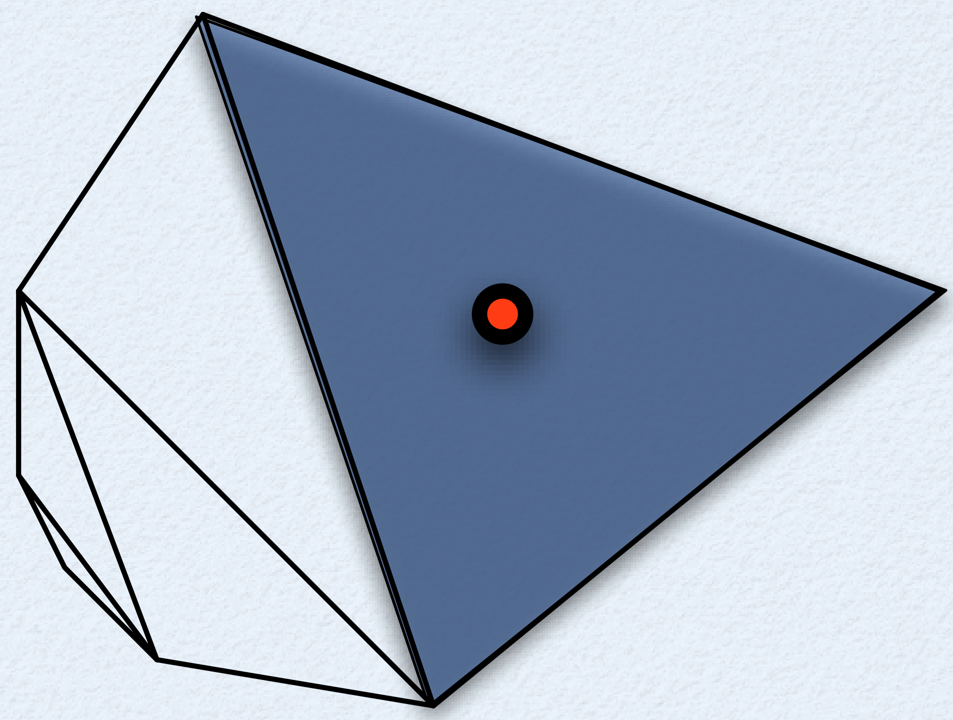
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

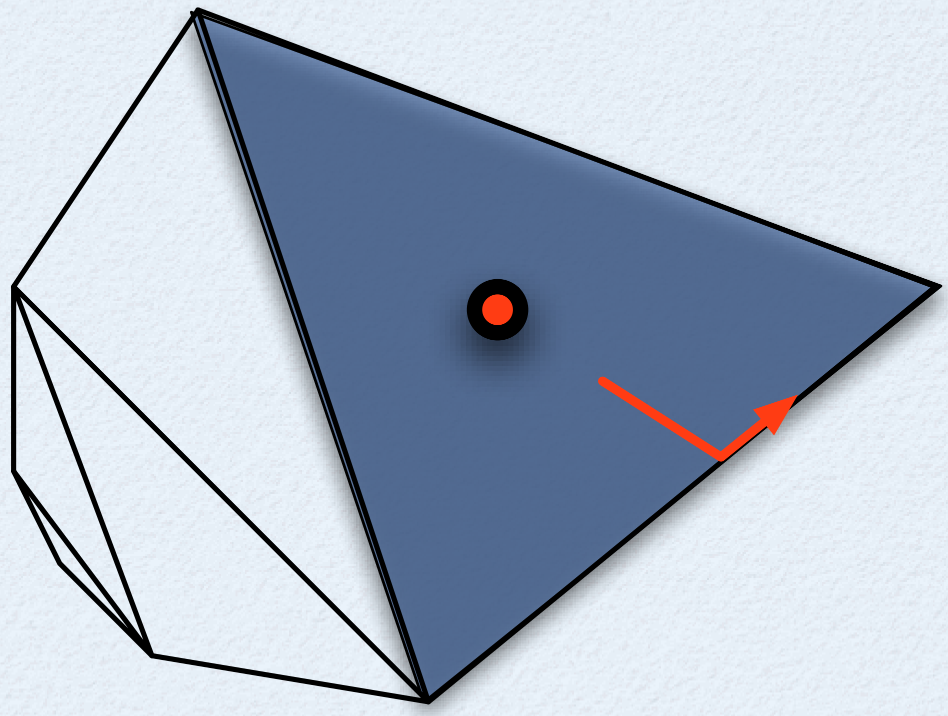
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

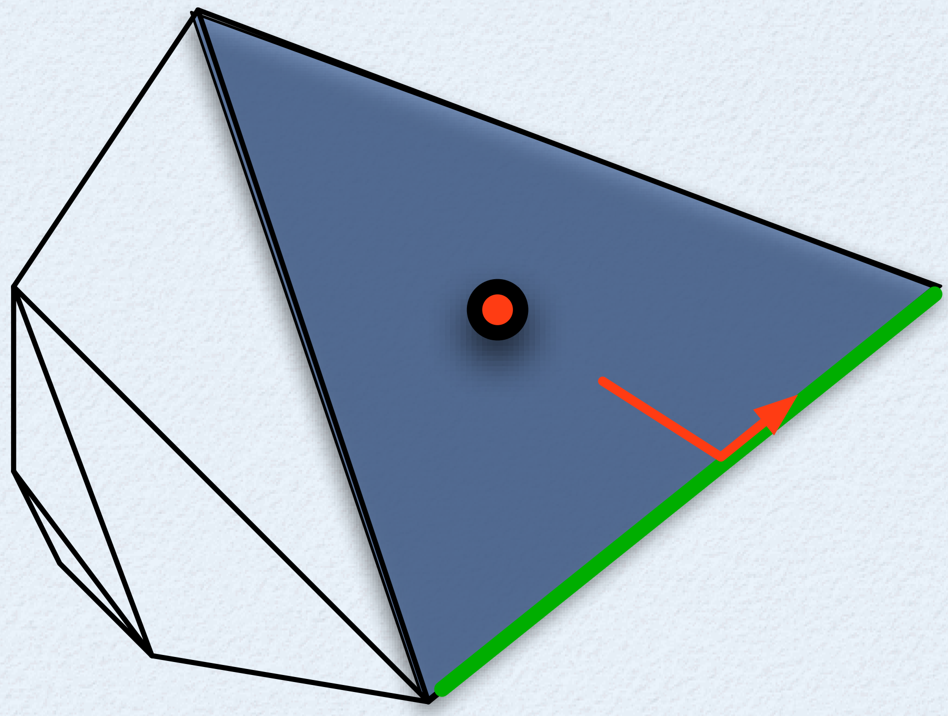
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

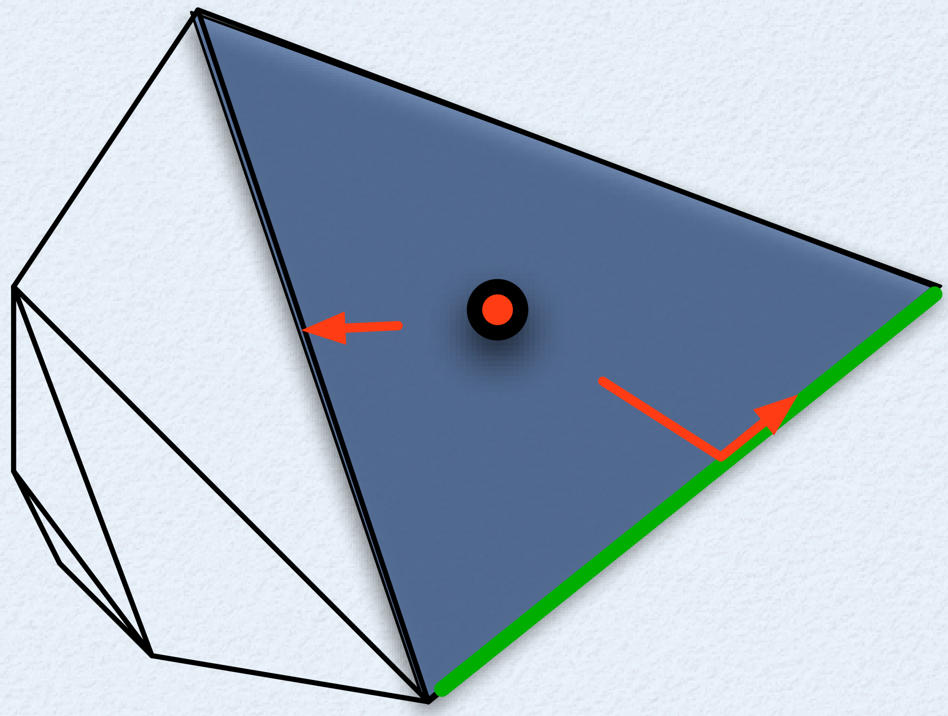
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

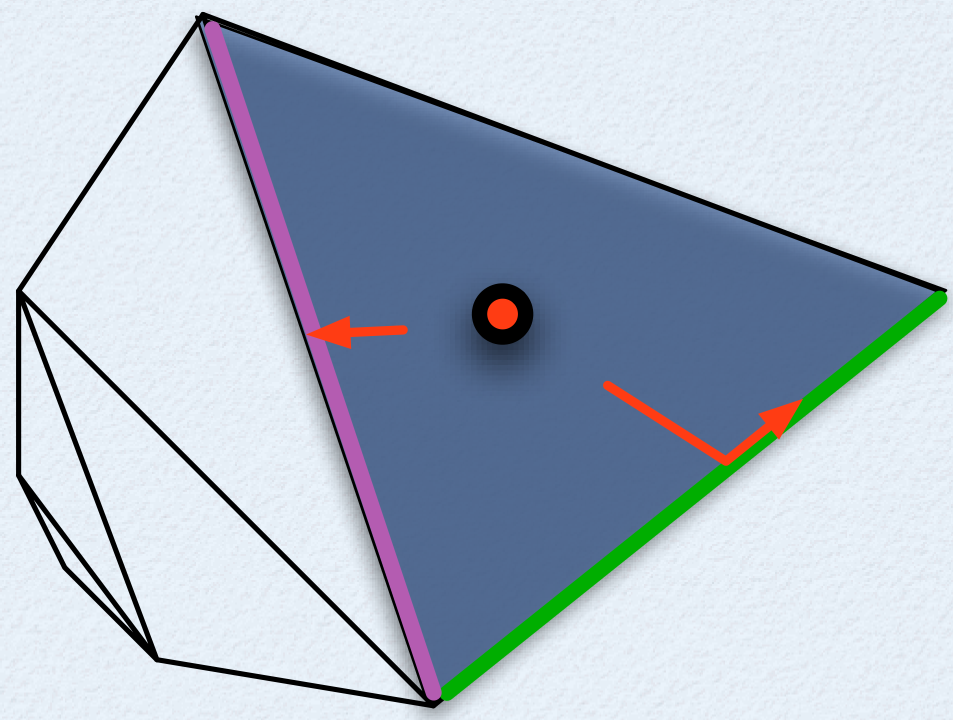
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

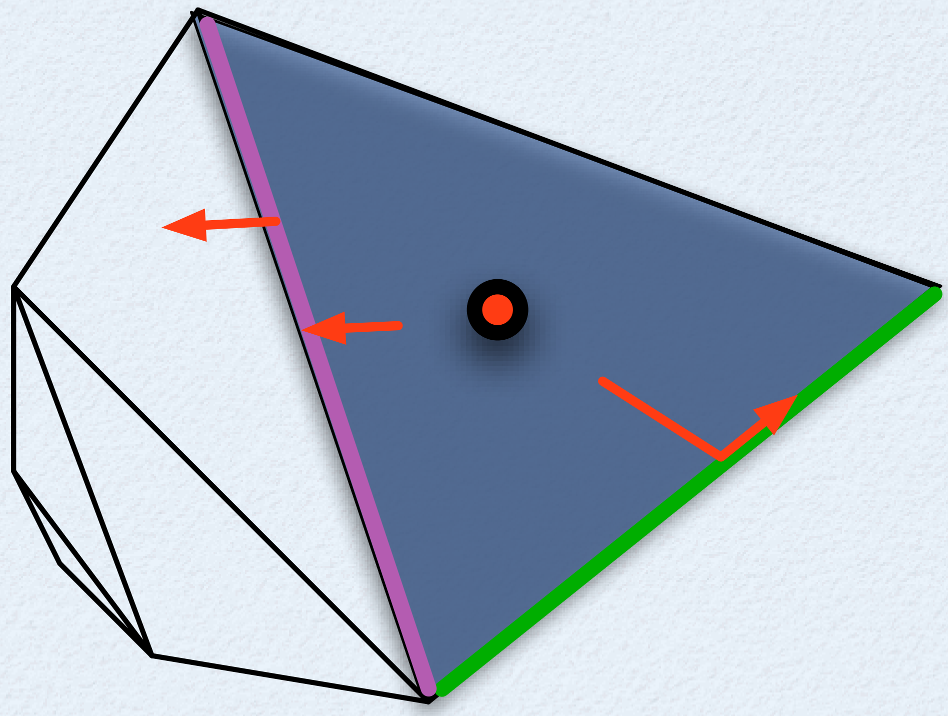
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

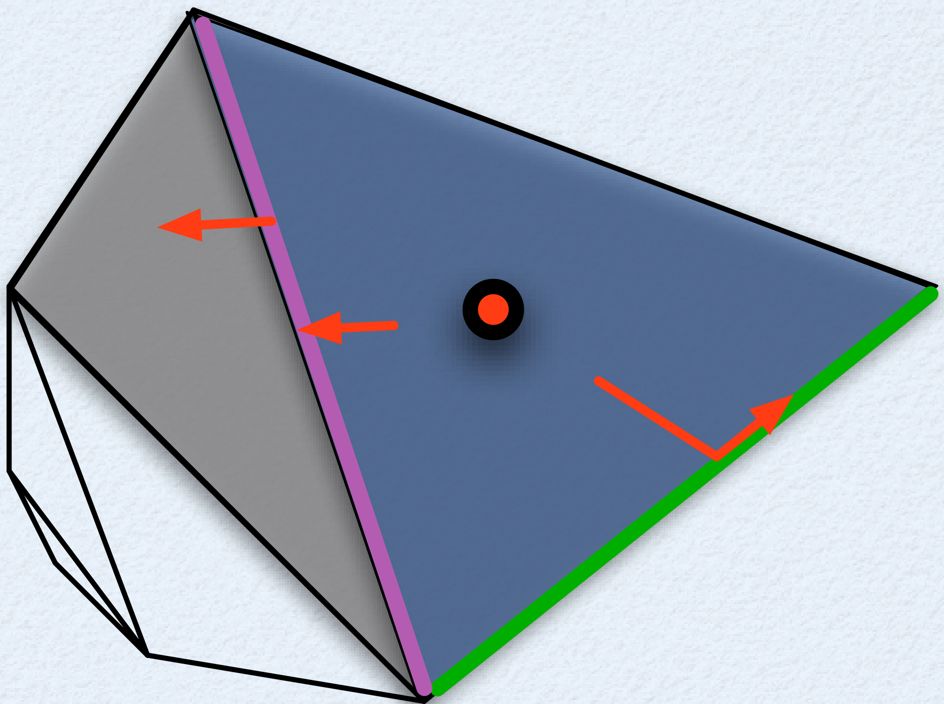
$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.



$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

A Partial Order on Delaunay Simplices

$\tau \prec \sigma$: some flow line of ϕ^* visits relative interiors of σ and τ consecutively.

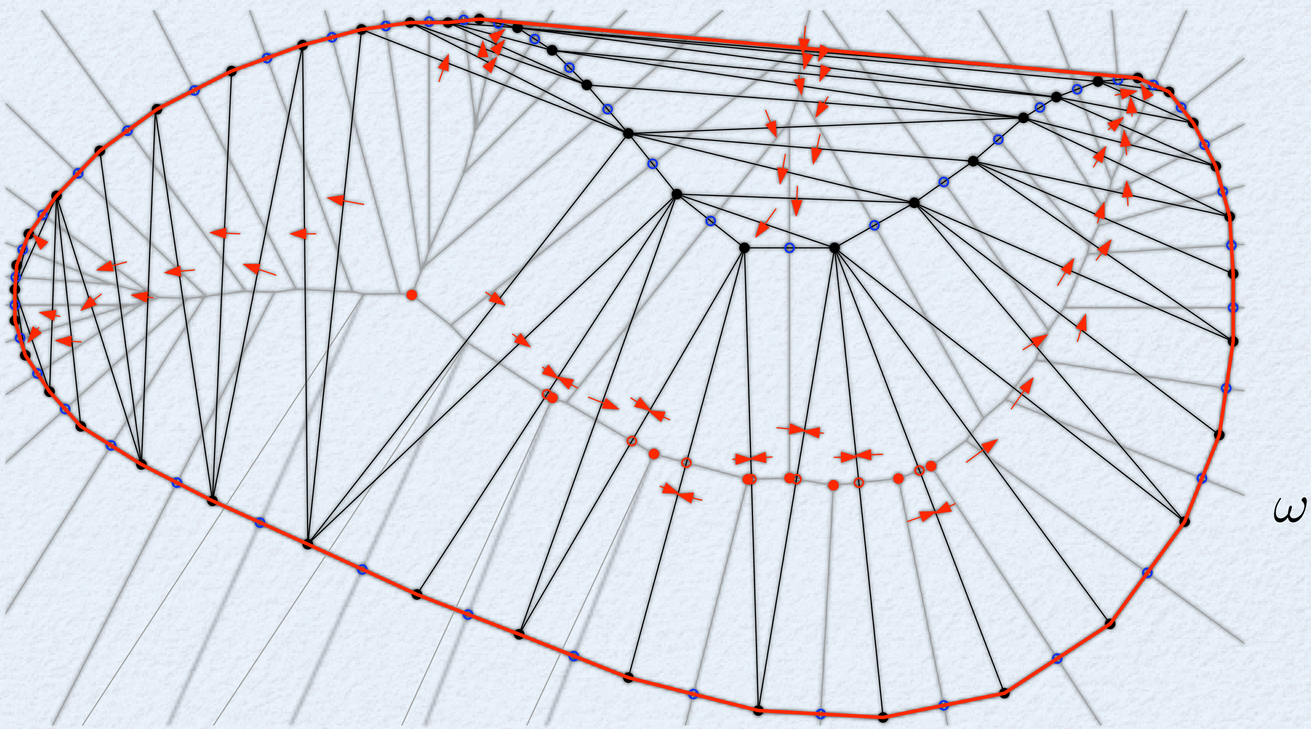


$\tau \prec^* \sigma$: there is a sequence $\tau = \tau_0 \prec \cdots \prec \tau_k = \sigma$.

The WRAP Algorithm [Edelsbrunner'04]

1. For every $\tau \in \mathrm{Del}\ P$, if the only critical simplex that precedes τ is the abstract critical simplex ω, then remove τ.

2. Return what is left as WRAP.

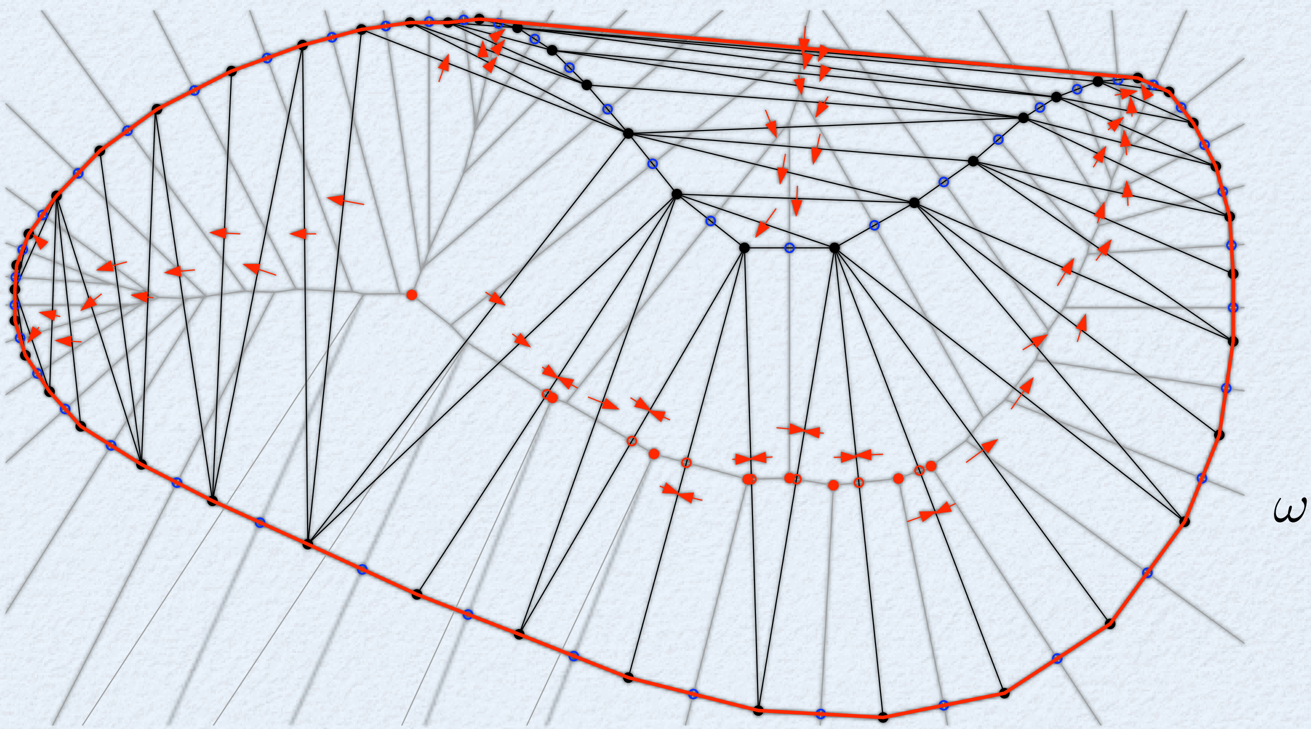


[Edelsbrunner'04] H. Edelsbrunner, Surface reconstruction by wrapping finite point-sets in space. Discrete & Computational Geometry, 2004.

The WRAP Algorithm [Edelsbrunner'04]

1. For every $\tau \in \text{Del } P$, if "every" critical simplex that precedes τ is an outer medial axis critical simplex, then remove τ.

2. Return what is left as WRAP.



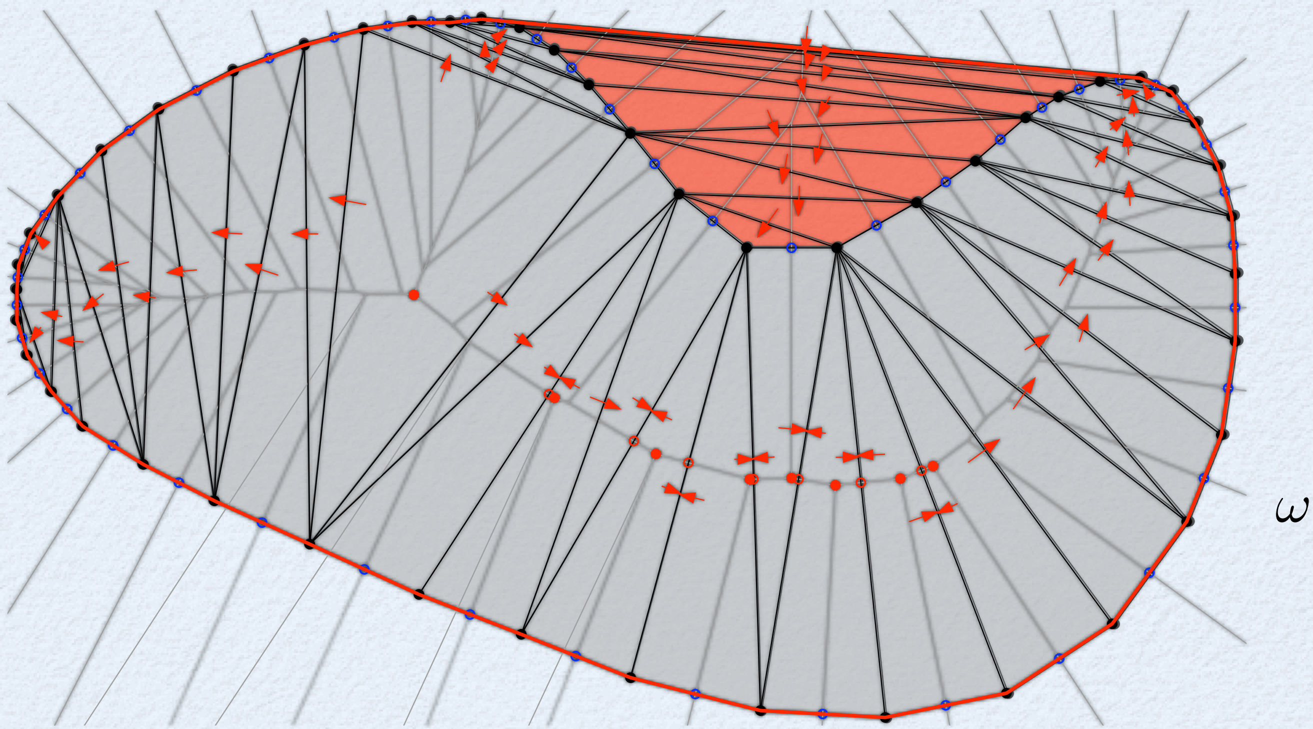
[Edelsbrunner'04] H. Edelsbrunner, Surface reconstruction by wrapping finite point-sets in space. Discrete & Computational Geometry, 2004.

The WRAP Algorithm

The WRAP Algorithm [Edelsbrunner'04]

1. For every $\tau \in \mathrm{Del}\ P$, if "every" critical simplex that precedes τ is an outer medial axis critical simplex, then remove τ.

2. Return what is left as WRAP.



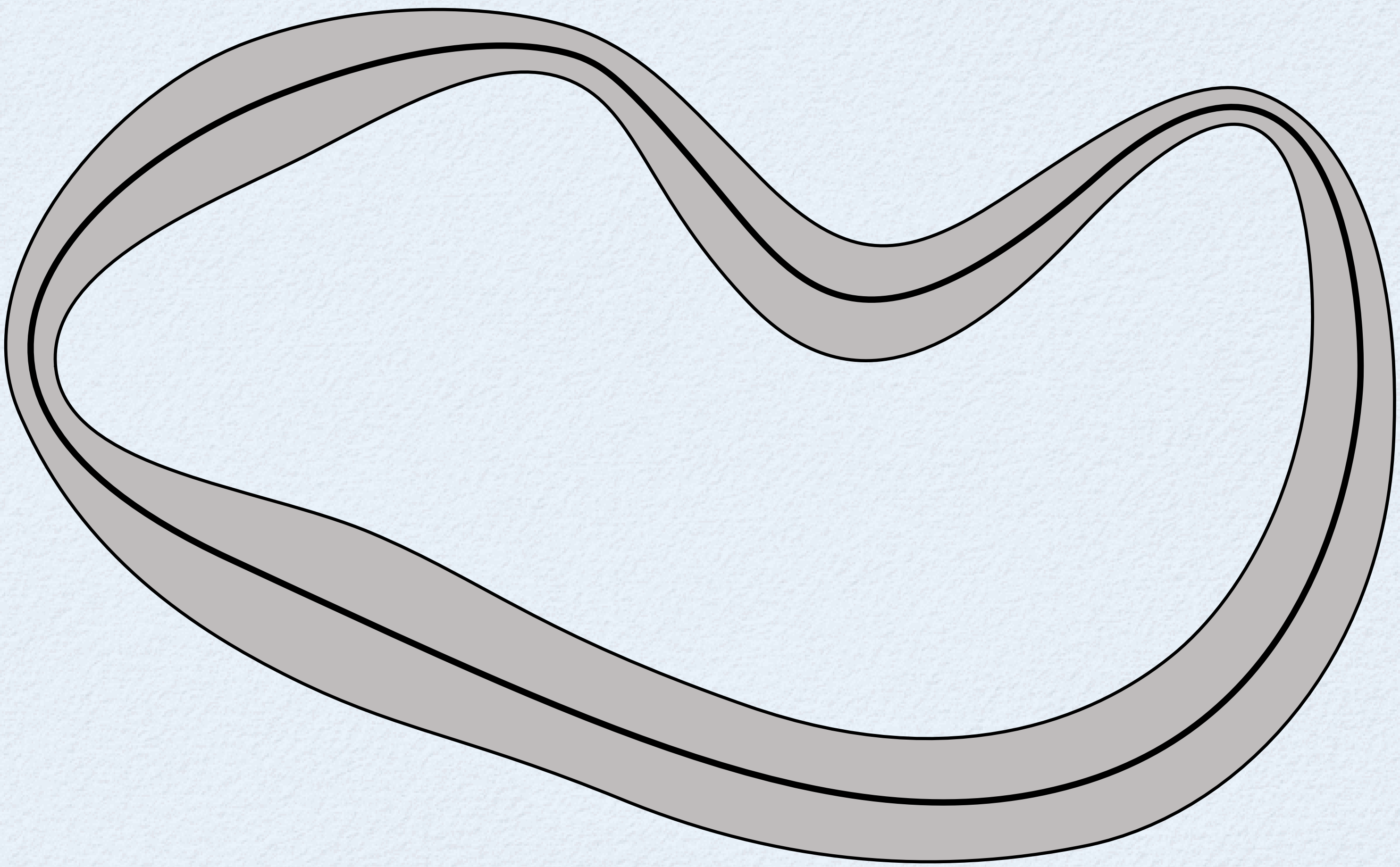
[Edelsbrunner'04] H. Edelsbrunner, Surface reconstruction by wrapping finite point-sets in space. Discrete & Computational Geometry, 2004.

The Guarantees

Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

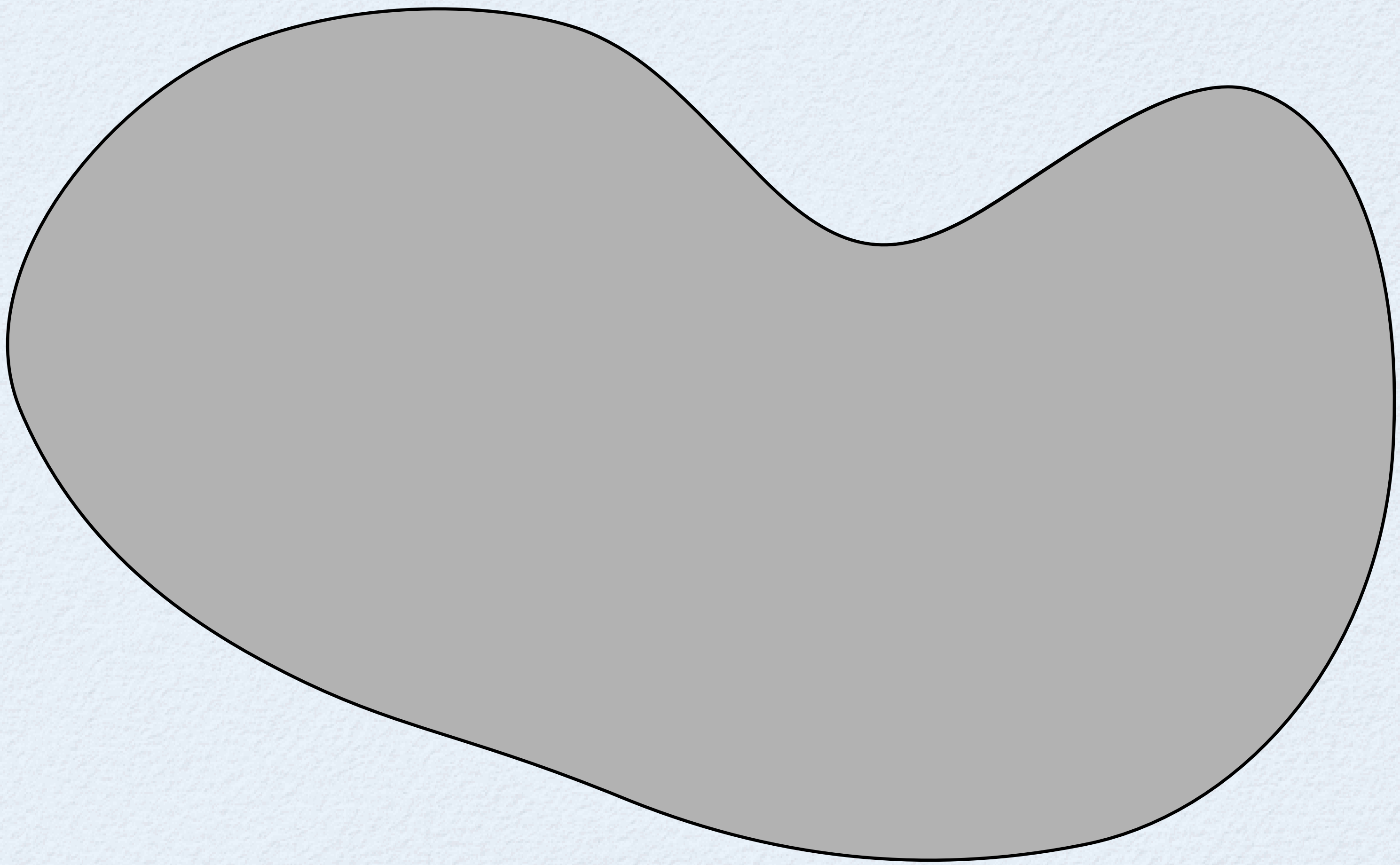
Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.



Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

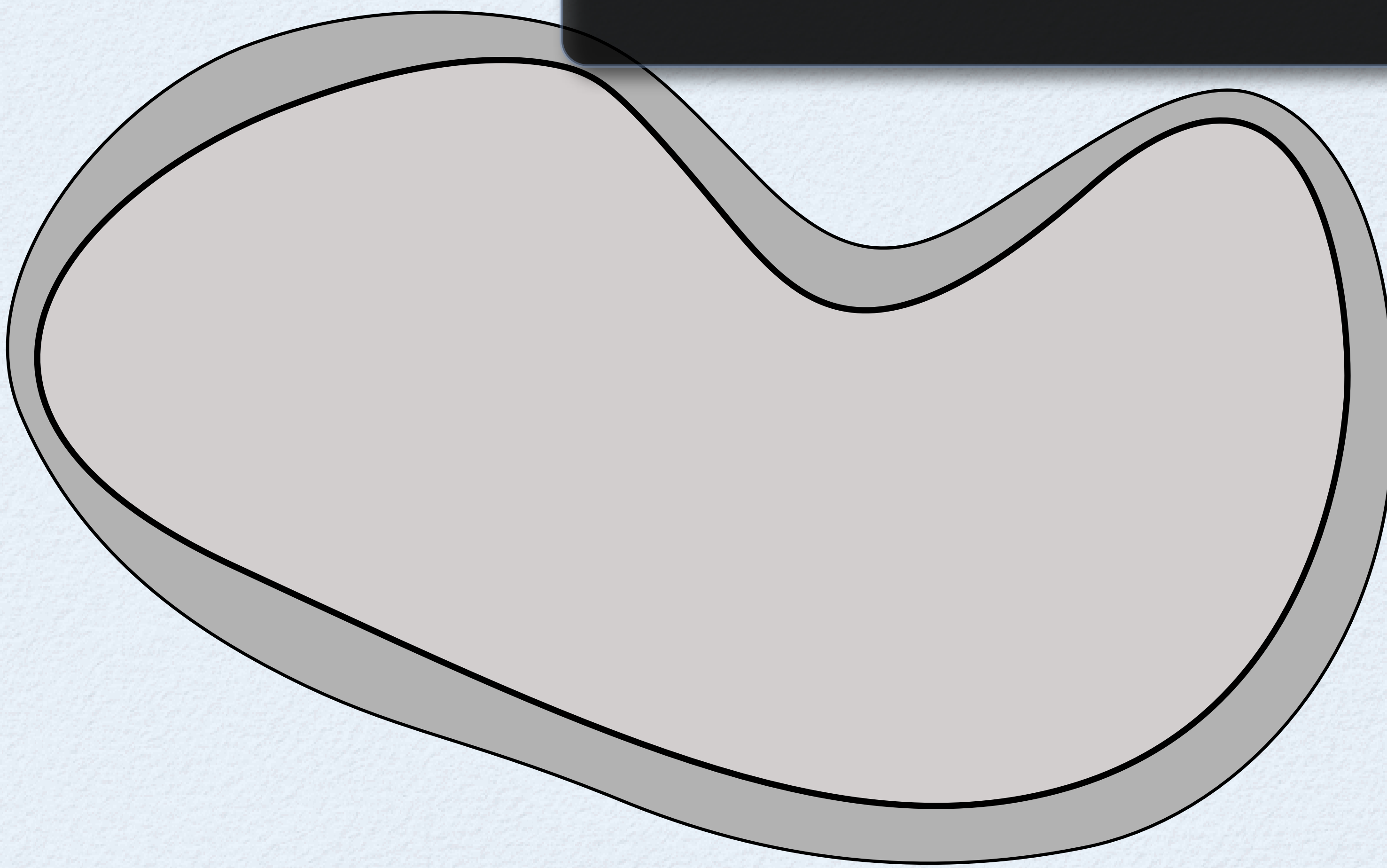


Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$

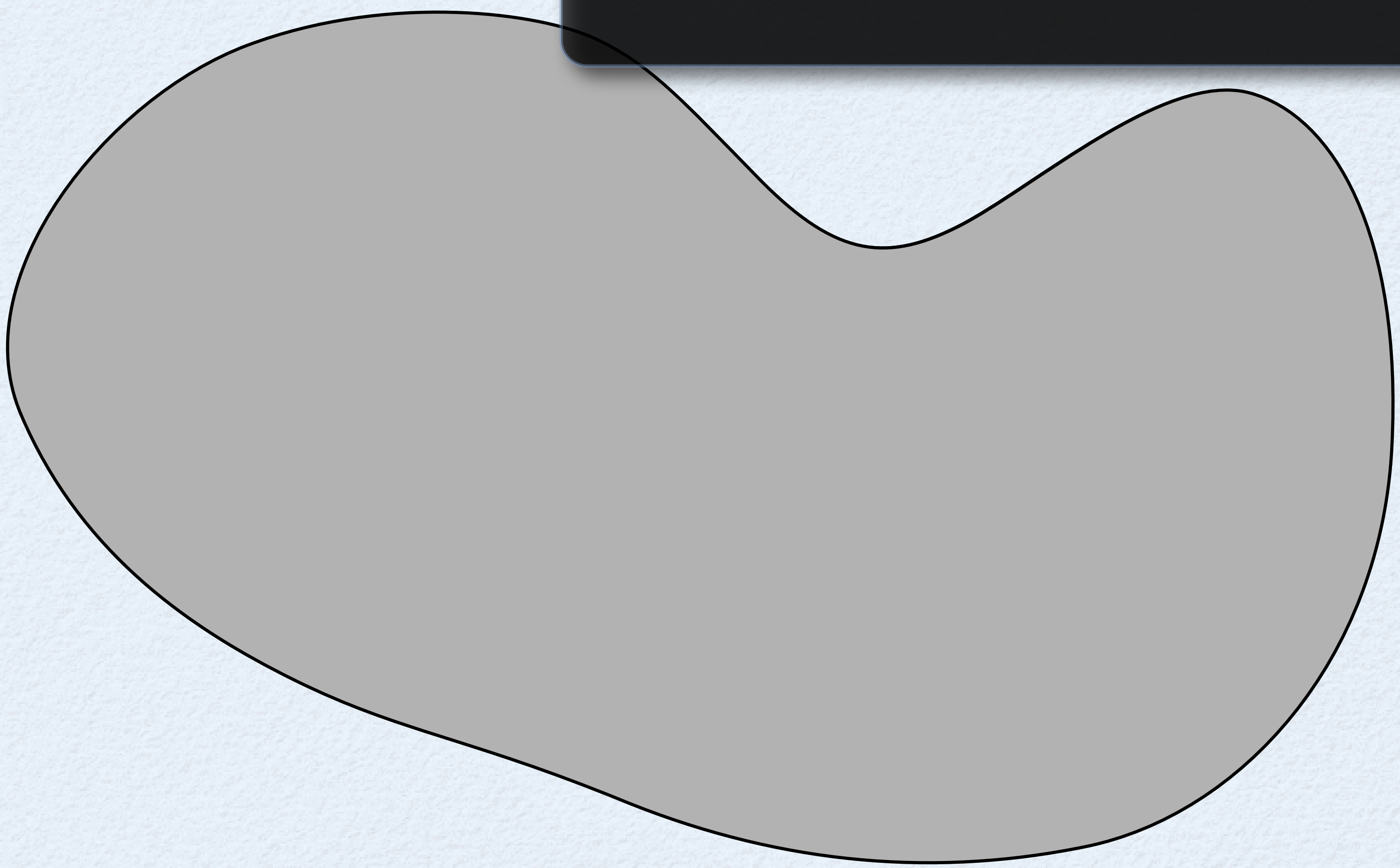


Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$

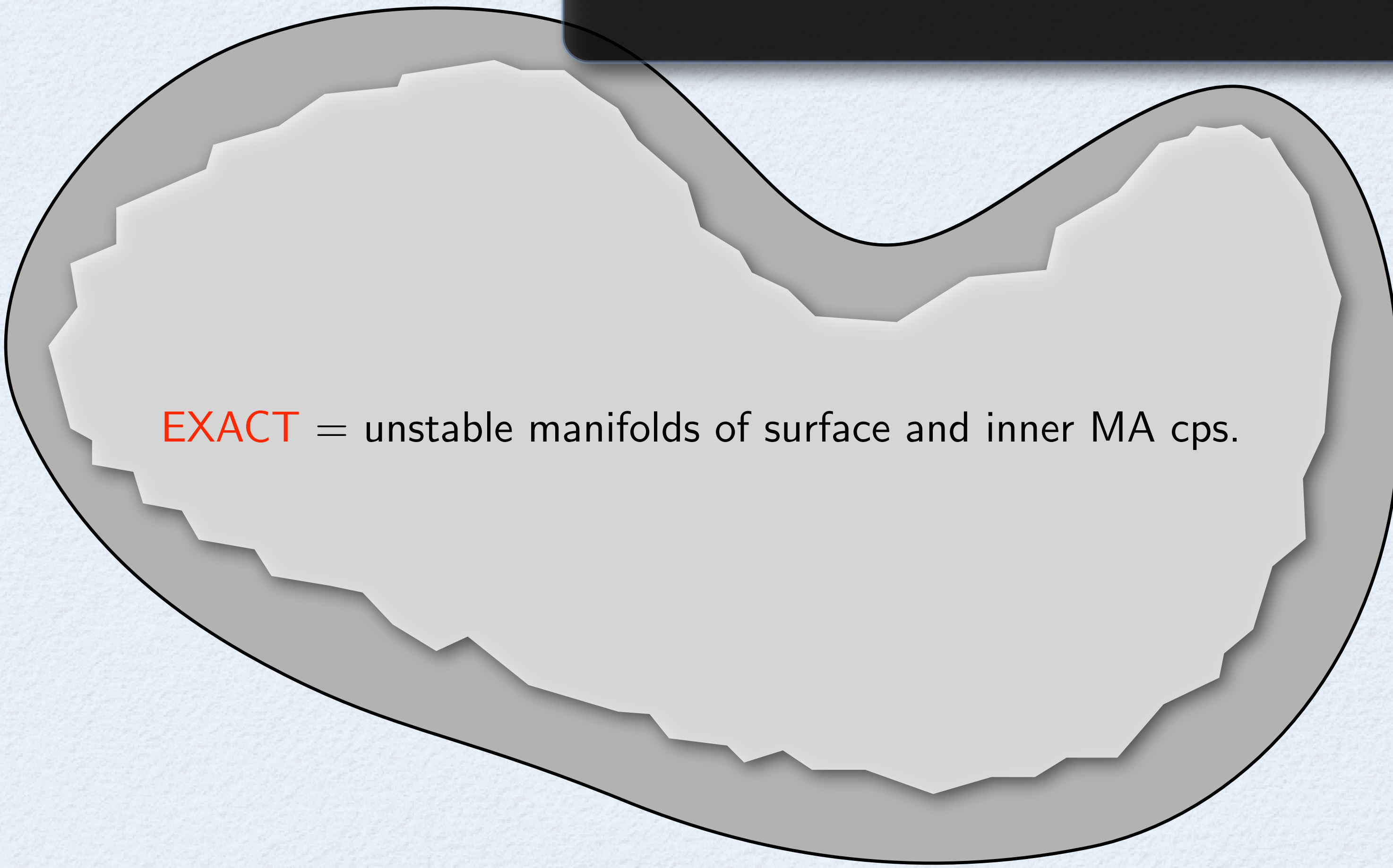


Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$



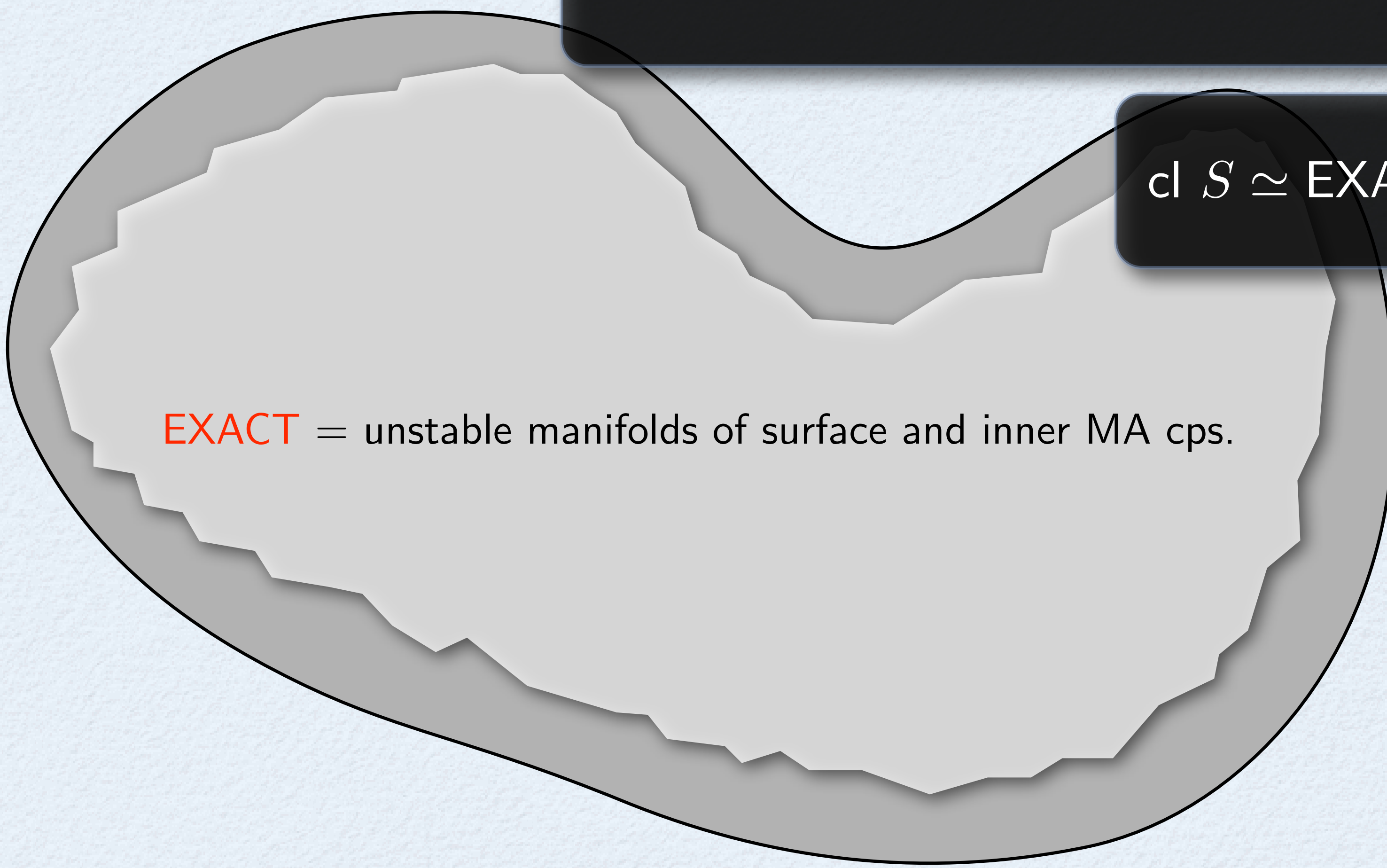
Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$

cl $S \simeq$ EXACT



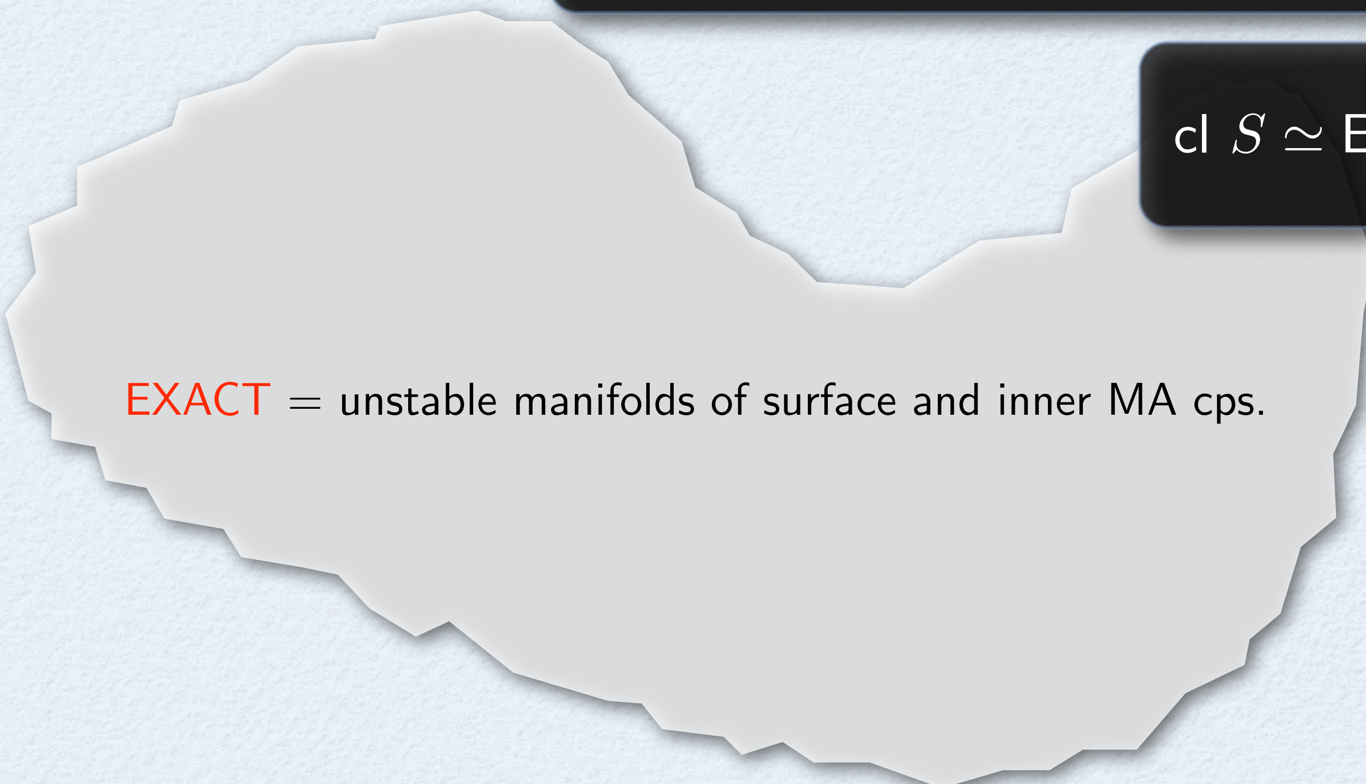
Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$

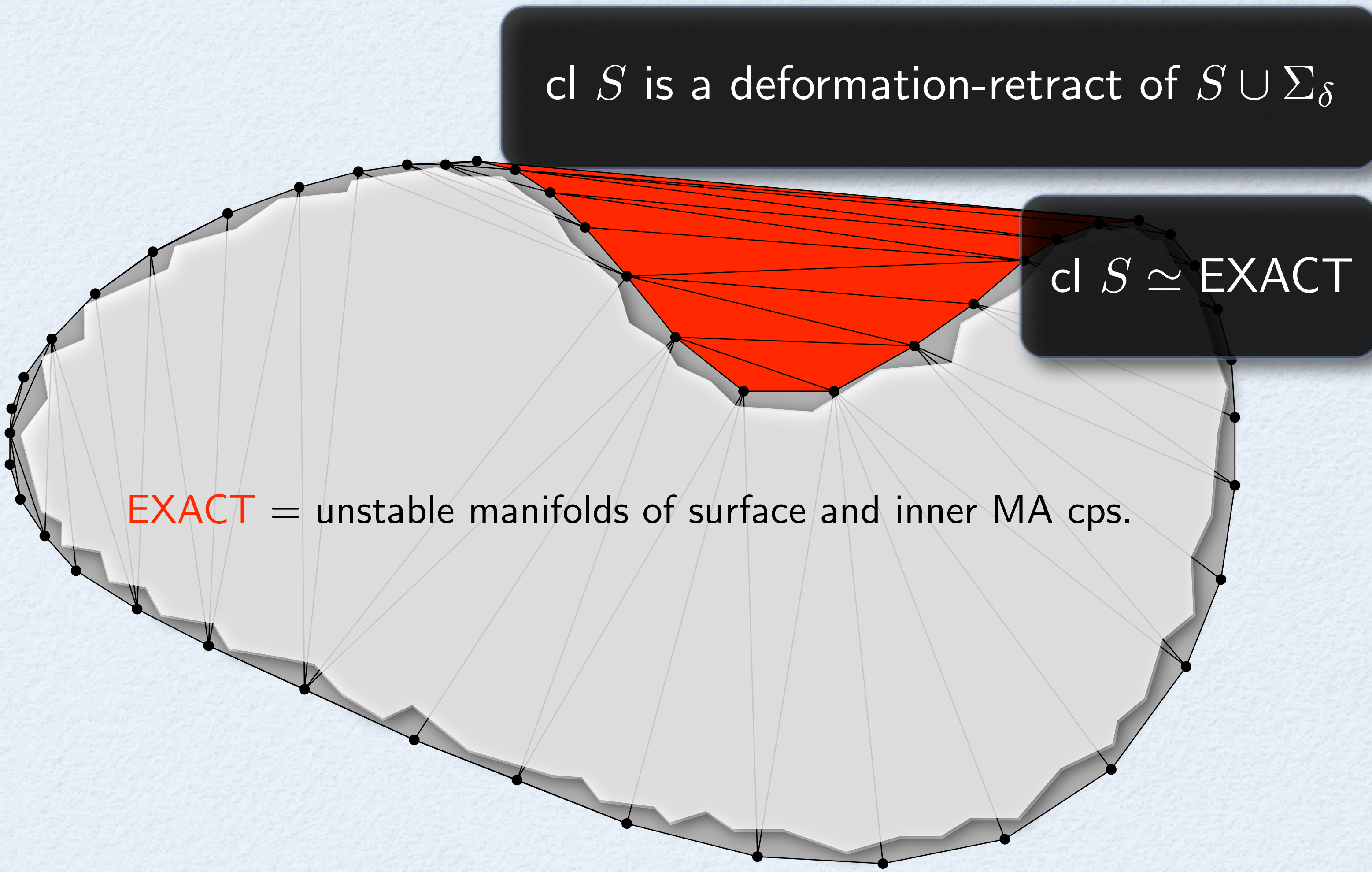
cl $S \simeq$ EXACT



Analysis of WRAP

Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.



Analysis of WRAP

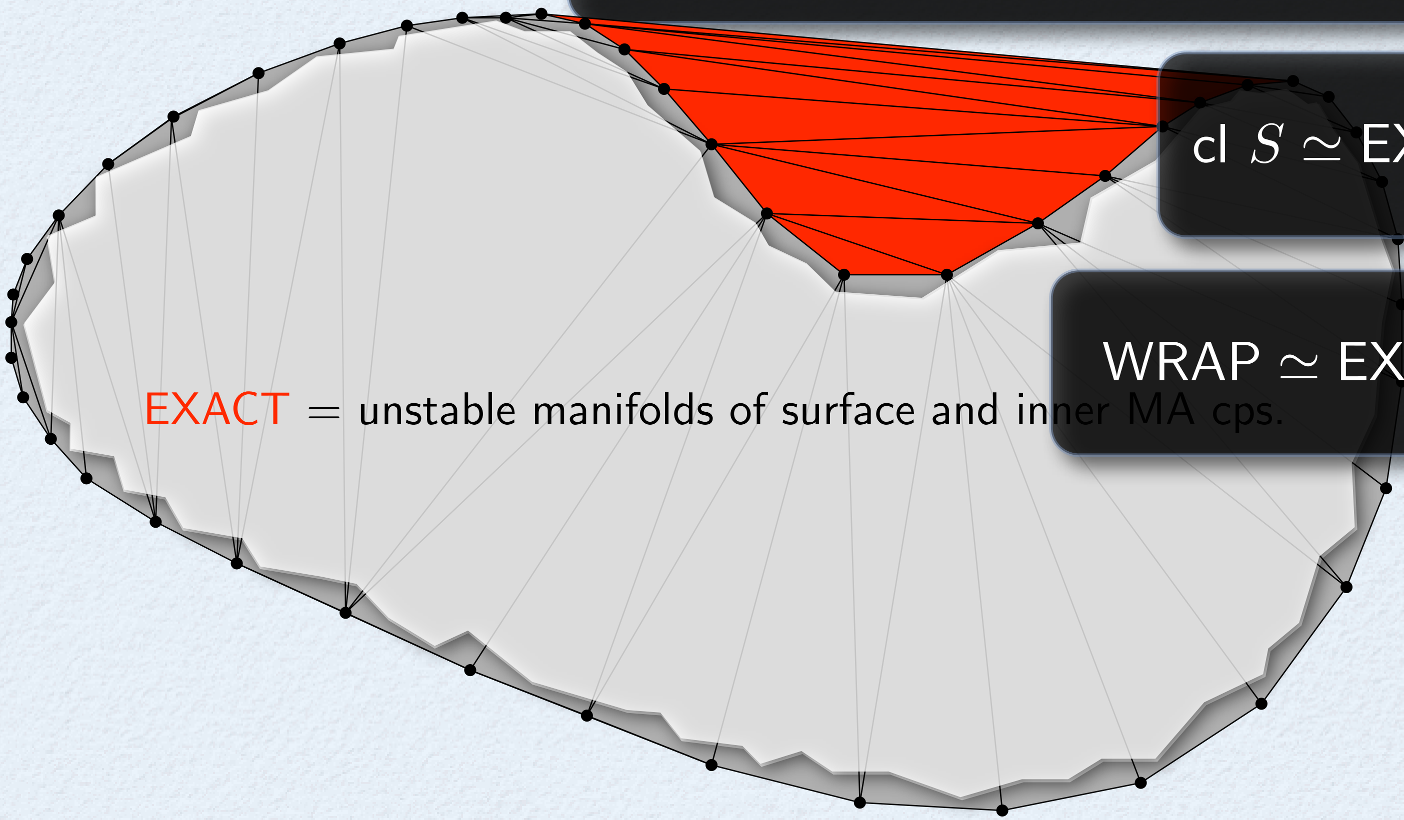
Theorem. WRAP and cl S are homotopy equivalent.

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$

cl $S \simeq$ EXACT

WRAP $\simeq$ EXACT



EXACT $=$ unstable manifolds of surface and inner MA cps.

Analysis of WRAP

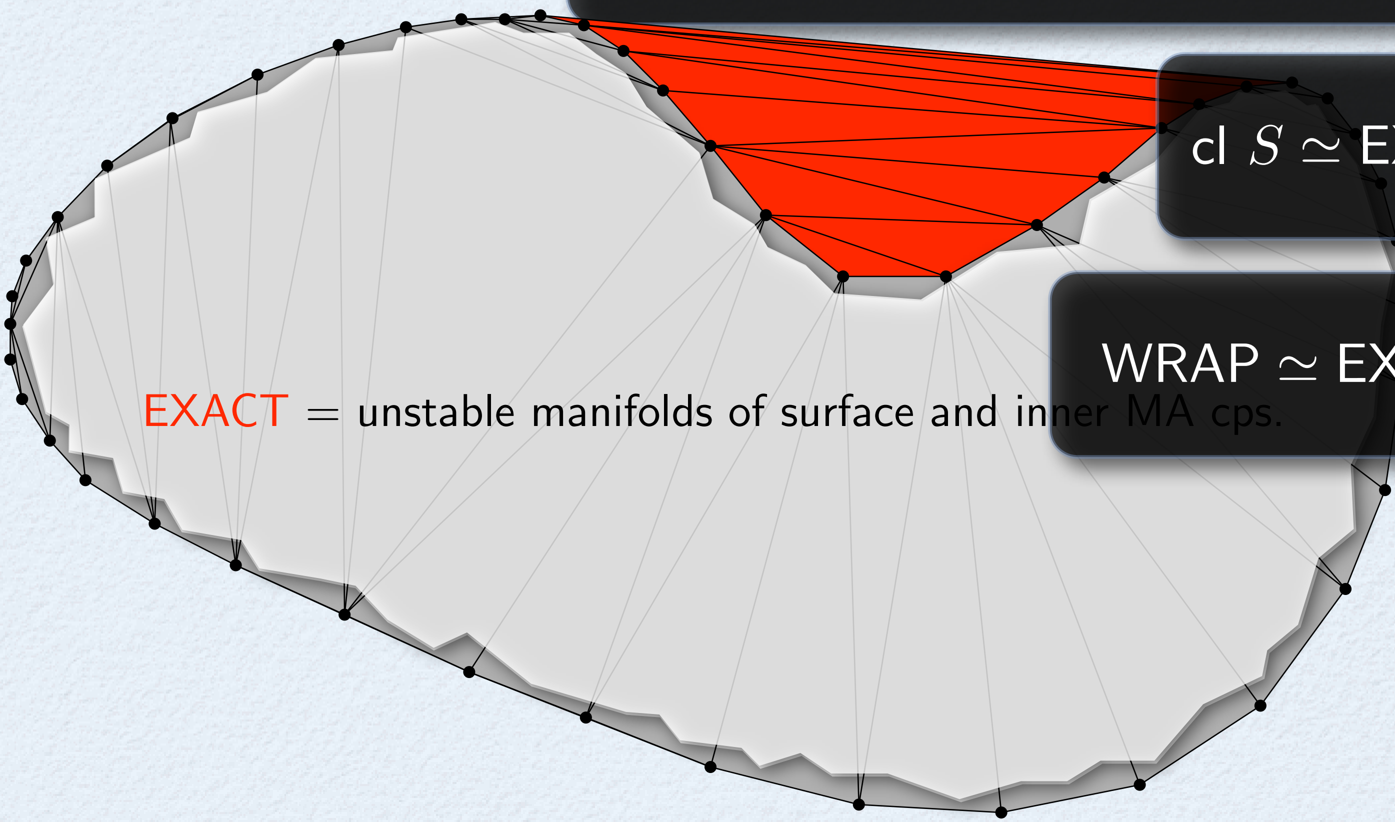
Theorem. WRAP and cl S are homotopy equivalent. (in 3D)

Lemma. $\phi^*(\Sigma_\delta) = \Sigma_\delta$.

cl S is a deformation-retract of $S \cup \Sigma_\delta$

cl $S \simeq$ EXACT

WRAP $\simeq$ EXACT



EXACT $=$ unstable manifolds of surface and inner MA cps.

Some Open Questions

- Can the geometric guarantee (and therefore the topological one) be extended to higher dimensions?

- Can WRAP be generalized for reconstruction of shapes with non-smooth boundaries? How should the sampling condition be defined? (some work done in [Lieutier-Chazal'06])

Some Open Questions

- Can the geometric guarantee (and therefore the topological one) be extended to higher dimensions?
- Can WRAP be generalized for reconstruction of shapes with non-smooth boundaries? How should the sampling condition be defined? (some work done in [Lieutier-Chazal'06])

Thank You!